



**SEP**  
SECRETARÍA DE  
EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

Instituto Tecnológico de  
Hermosillo

## DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

Sistema para el Control Domótico como Soporte a  
Personas con Esclerosis Lateral Amiotrófica (ELA)

# TESIS

Presentada por:  
Ing. Roberto Lorenzo Aguilar Maldonado

Como requisito parcial para obtener el grado de  
Maestro en Ciencias de la Computación

Director de tesis:  
M.C. César Enrique Rose Gómez

Hermosillo, Sonora, México



Av. Tecnológico S/N Col. El Sahuaro C.P. 83170 Hermosillo, Sonora  
Tel. 01 (662) 2-606500, Ext. 111, 177 e-mail: [plan\\_hermosillo@tecnm.mx](mailto:plan_hermosillo@tecnm.mx)

[www.tecnm.mx](http://www.tecnm.mx) | [www.ith.mx](http://www.ith.mx)



**ISO 9001:2015**  
Sistema de Gestión de Calidad Certificado



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

Instituto Tecnológico de Hermosillo  
División de Estudios de Posgrado e Investigación

SECCIÓN: DIV. EST. POS. E INV.  
No. OFICIO: DEPI/121/21  
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN  
DE TESIS.

25 de Enero de 2021

**C. ROBERTO LORENZO AGUILAR MALDONADO,  
PRESENTE.**

Por este conducto, y en virtud de haber concluido la revisión del trabajo de tesis que lleva por nombre "**SISTEMA PARA EL CONTROL DOMÓTICO COMO SOPORTE A PERSONAS CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)**", que presenta para el examen de grado de la MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN, y habiéndola encontrado satisfactoria, nos permitimos comunicarle que se autoriza la impresión del mismo a efecto de que proceda el trámite de obtención de grado.

Deseándole éxito en su vida profesional, quedo de usted.

ATENTAMENTE

M.C. CÉSAR ENRIQUE ROSE GÓMEZ  
DIRECTOR

DRA. MARÍA TRINIDAD SERNA ENCINAS  
SECRETARIA

M.C. RAFAEL ARMANDO GALAZ BUSTAMANTE  
VOCAL

M.C.O. ROSA IRENE SÁNCHEZ FERMÍN  
JEFA DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN



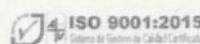
S.E.P.

RISF/fjcg\*

INSTITUTO TECNOLÓGICO  
DE HERMOSILLO  
DIVISIÓN DE ESTUDIOS  
DE POSGRADO



Av. Tecnológico S/N Col. El Sahuaro C.P. 83170 Hermosillo, Sonora  
Tel. 01 (662) 260 65 00, ext. 136, e-mail: [depi\\_hermosillo@tecnm.mx](mailto:depi_hermosillo@tecnm.mx)  
[tecnm.mx](http://tecnm.mx) | [ith.mx](http://ith.mx)



# Resumen

Actualmente, se estima que en una de cada 20,000 personas en el mundo es diagnosticada con Esclerosis lateral amiotrófica y en México aproximadamente se encuentran 6,000 casos con este padecimiento, esta enfermedad no cuenta con cura y se desconocen las causas que la originan, este padecimiento afecta las motoneuronas que son las células encargadas de llevar a cabo el movimiento de los músculos del cuerpo, es una enfermedad degenerativa que comienza con caídas repentinas y el llevar a cabo ciertos movimientos como el abotonamiento de una camisa, en casos graves se manifiesta con la pérdida de habilidad de deglutir alimentos así como la habilidad de respirar de forma autónoma, sin embargo no se ve afectado ningún aspecto cognitivo ni nada relacionado con el movimiento de la mirada.

Este proyecto se enfoca en el desarrollo de un sistema tipo tableta para llevar a cabo el control de las cosas del hogar y la comunicación mediante mensajería por medio de un sensor de seguimiento de mirada teniendo como referencia personas con esclerosis lateral amiotrófica.

La metodología que se siguió para el desarrollo del sistema es la del análisis y diseño orientado a objetos y se utilizó UML, el lenguaje de modelado unificado, para ayudar a identificar los componentes del sistema de acuerdo con el modelo de arquitectura 4+1 vistas. A partir de los elementos identificados se desarrolló la arquitectura y los diagramas correspondientes para describir los procesos que integran el sistema.

Al Concluir se llevó a cabo el diseño e implementación de una interfaz gráfica usable a través del seguimiento de la mirada, la cual consta de tres módulos: iluminación (Encendido y apagado de un foco mediante el seguimiento de mirada), Comunicación (Mandar mensajes a través de aplicaciones de mensajería instantánea para teléfonos inteligentes y SMS a través de internet, mediante el seguimiento de mirada), Tablero (Comunicación por parlantes de la tableta mediante el seguimiento de mirada) y también el uso de otros componentes del hogar mediante el replicado de controles con tecnología por infrarrojo.

# Dedicatoria

Antes de nada, agradezco infinitamente a Dios por darme salud y vida para poder hacer realidad esta tesis y hacerme sentir especial premiándome con mi familia, ya que sin ellos no hubiera podido lograr este gran paso en mi vida.

Estaré siempre agradecido con mis abuelos Aurelia Sanora, Emma Murrieta y Mario Maldonado por su incondicional e inquebrantable apoyo, espero y puedan encontrar un poco de amor y reconocimiento en estas breves palabras.

A mis padres Roberto Aguilar e Ivette Maldonado por estar siempre presentes en todos los aspectos de mi vida, por brindarme su apoyo en cada paso e inculcarme buenos valores, por tantos consejos, amor y cariño.

A mi hermano Aarón Aguilar por los buenos momentos compartidos, a Karen Ruiz por darme la gran bendición de ser padre de mi hijo Santiago Aguilar.

Valoro mucho todo lo que han hecho por mí y lo que me han dado, gracias por hacerme creer que puedo lograr todo lo que me proponga, por sus oraciones, bendiciones y amor sincero, soy la persona más afortunada por tenerlos como familia, siempre estaré en infinita deuda con ustedes, gracias.

# Agradecimientos

A mi director de tesis, M.C. César Enrique Rose Gómez, por la confianza brindada durante estos últimos años para la realización de este proyecto, por todo el apoyo, buena comunicación y enseñanzas; Muchas gracias por poder contar con usted en todo momento.

A mi ex-director de tesis, M.C. Ana Luisa Millán Castro, por la confianza depositada en mí para la realización de esta investigación, por siempre contar con su apoyo, inspiración fundamental por la cual querer ser Maestro en Ciencias de la computación, gracias por su valioso tiempo y todo el conocimiento compartido, los cuales fueron esenciales para cumplir los objetivos propuestos.

A mi M.C. Sonia Regina Meneses, por darme siempre un consejo y recibirme de buena manera, por orientarme en todos los aspectos y brindarme palabras de aliento. A mis sinodales, Dra. María Trinidad Serna Encinas y M.C. Rafael Armando Galaz Bustamante, por el valioso tiempo que dedicaron en la revisión de este trabajo de tesis y sus comentarios.

A todos mis compañeros de generación por toda la ayuda y buenos momentos compartidos dentro y fuera del laboratorio.

Al núcleo académico de la Maestría en Ciencias de la Computación del Instituto Tecnológico de Hermosillo, por su apoyo incondicional, su trato cordial y accesible en todo momento. Gracias por su invaluable labor de enseñanza y por fomentar el aprendizaje.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por su apoyo económico mediante la beca de manutención otorgada durante 2 años.

A todos les estaré eternamente agradecido por formar parte de esta maravillosa etapa que me permitido ser un mejor profesional y crecer como persona.

# Índice

Lista de ilustraciones .....	8
1 Introducción .....	10
1.1 Antecedentes .....	11
1.2 Planteamiento del problema.....	12
1.2.1 Objetivos (generales y específicos) .....	13
1.2.2 Objetivo Específico .....	13
1.3 Objetivo General.....	13
1.4 Justificación de la tesis .....	13
1.5 Alcances y delimitaciones .....	14
1.6 Metodología.....	14
1.7 Organización de la tesis .....	15
2 Estado del arte .....	16
2.1 Introducción.....	16
2.2 ELA (esclerosis lateral amiotrófica) .....	16
2.3 Domótica.....	18
2.4 Sistemas Embebidos .....	19
2.5 Sensores .....	20
2.6 Tipo De Sensores .....	20
2.7 Actuadores .....	21
2.8 Comunicación embebida.....	22
2.9 Módulos de radio frecuencia.....	23
2.10 Módulo ESP8266 .....	24
2.11 Módulo ESP32 .....	25
2.12 Módulo XBEE.....	26
2.13 Módulos Comunicación Celular.....	27
2.14 Coordinador domótico.....	28
2.15 Prototipado de embebidos .....	32
2.16 Sensores de seguimiento de ojos.....	34
2.17 Librerías Seguimiento Ocular .....	36
2.18 Servicios Web .....	37

2.19	Trabajos relacionados.....	40
3	Análisis y diseño del sistema.....	43
3.1	Introducción.....	43
3.2	Arquitectura del sistema.....	43
3.3	Diseño del Sistema.....	44
3.3.1	Diagrama de vista de procesos.....	45
3.3.2	Diagrama de vista Lógica.....	46
3.3.3	Diagrama de implementación.....	47
3.3.4	Diagrama de vista de despliegue.....	47
3.3.5	Diagrama de vista de escenarios.....	48
3.3.6	Especificaciones de casos de uso.....	49
3.4	Diseño de la base de datos.....	52
3.4.1	Diseño conceptual (Modelo ER).....	52
3.4.2	Transformación al modelo Relacional.....	52
3.4.3	Normalización.....	53
3.4.4	Implementación de la base de datos.....	54
4	Implementación del sistema.....	56
4.1	Interfaz del sistema.....	57
4.2	Triangulación de comunicación.....	65
4.3	Control Iluminación.....	68
4.4	Envío de mensajería instantánea.....	71
4.5	Sintetizador voz por teclado.....	72
5	Prueba y Análisis de resultados.....	73
5.1	Configuraciones Iniciales.....	73
5.2	Modo Nocturno.....	74
5.3	Módulo Domótico Encendido Luz.....	78
5.4	Módulo Comunicación.....	78
5.5	Módulo Tablero.....	78
5.6	Pruebas Televisión.....	79
5.7	Validación del Sistema.....	83
6	Conclusiones y trabajos futuros.....	85
6.1	Conclusiones.....	85
6.2	Trabajos Futuros.....	85
	Bibliografía.....	89

# Lista de ilustraciones

Ilustración 2.1 Sistema embebido.....	20
Ilustración 2.2 Sensores, aspecto físico.....	21
Ilustración 2.3 Tipos de actuadores .....	22
Ilustración 2.4 Módulos de comunicación.....	24
Ilustración 2.5 ESP8266 .....	25
Ilustración 2.6 ESP32 .....	26
Ilustración 2.7 Módulos xbee .....	27
Ilustración 2.8 XBEE Celular Modelo XB3-C-A2-UT-001.....	28
Ilustración 2.9 Raspberry-pi 4 .....	29
Ilustración 2.10 UWP multiplataforma .....	30
Ilustración 2.11 Windows IOT dashboard.....	31
Ilustración 2.12 Windows 10 raspberry.....	32
Ilustración 2.13 Arduino.....	32
Ilustración 2.14 Adaptaciones Arduino .....	33
Ilustración 2.15 Reflexión Luz Infrarroja SO.....	34
Ilustración 2.16 IrisBound .....	35
Ilustración 2.17 TM5 mini.....	35
Ilustración 2.18 Tobii eye tracker 4C .....	35
Ilustración 2.19 Permisos de periféricos.....	36
Ilustración 2.20 Calibración WebGaze.....	37
Ilustración 2.21Recepción Mensajería NiceApi.net.....	39
Ilustración 2.22 Kit Inteligente WYZE .....	40
Ilustración 2.23 Nest Hub.....	41
Ilustración 2.24 Alexa echo studio .....	41
Ilustración 2.25 Asistente virtual.....	42
Ilustración 3.1 Arquitectura propuesta .....	43
Ilustración 3.2 Diagrama de secuencia .....	46
Ilustración 3.3 Diagrama de vista lógica .....	46
Ilustración 3.4 Diagrama de implementación.....	47
Ilustración 3.5 Diagrama de vista de despliegue .....	48
Ilustración 3.6 Diagrama de vista de escenarios.....	49
Ilustración 3.7 Diagrama entidad relación.....	52
Ilustración 3.8 Diagrama de base de datos .....	55
Ilustración 4.1 Gestor de perfil de usuarios.....	57
Ilustración 4.2 Calibración de dispositivo .....	58
Ilustración 4.3 Permisos seguimiento ocular .....	58
Ilustración 4.4 Manifiesto.....	59
Ilustración 4.5 Pantalla principal .....	59



Ilustración 4.6 Radial progress bar .....	60
Ilustración 4.7 estructura xaml vista principal.....	60
Ilustración 4.8 Variables globales .....	61
Ilustración 4.9 Inicialización seguimiento de mirada.....	61
Ilustración 4.10 Observador mirada .....	62
Ilustración 4.11 configuración Seguimiento.....	62
Ilustración 4.12 Estatus Dispositivo .....	62
Ilustración 4.13 Definición Elipse .....	63
Ilustración 4.14 Conteo Progreso .....	63
Ilustración 4.15 Temporizador de mirada.....	64
Ilustración 4.16 Despliegue de Menú.....	64
Ilustración 4.17 Petición al Coordinador.....	65
Ilustración 4.18 Ip estática Web Api .....	65
Ilustración 4.19 Web API Controller.....	66
Ilustración 4.20 Icono Encendido .....	66
Ilustración 4.21 API'S Mensajería de terceros .....	67
Ilustración 4.22 Configuración C# Mensajería .....	68
Ilustración 4.23 Sketch Arduino ESP826 .....	69
Ilustración 4.24 Sketch On && Off.....	70
Ilustración 4.25 Modulo Esp8266 .....	70
Ilustración 4.26 Interfaz Modulo Comunicación.....	71
Ilustración 4.27 Interfaz Tablero .....	72
Ilustración 4.28 Comunicación mediante parlantes en C# .net .....	73
Ilustración 5.1 Control ocular inicio Windows .....	74
Ilustración 5.2 Registro inicio sesión biométrico .....	75
Ilustración 5.3 Módulo Iluminación Apagado.....	76
Ilustración 5.4 Módulo Iluminación encendida.....	76
Ilustración 5.5 Módulo comunicación .....	77
Ilustración 5.6 Módulo tablero .....	77
Ilustración 5.7 Control Televisor.....	79
Ilustración 5.8 Módulo TV .....	80
Ilustración 5.9 Configuración Televisor.....	81
Ilustración 5.10 TV LG .....	82
Ilustración 5.11 Modelos Minisplit .....	82
Ilustración 6.1 Estetoscopio Digital .....	88

# 1 Introducción

La esclerosis lateral amiotrófica (ELA) es una enfermedad que afecta las neuronas encargadas de llevar a cabo los movimientos de los músculos, este tipo de neuronas se le conocen como motoneuronas o neuronas motoras. Una persona diagnosticada con ELA, manifiesta debilidad muscular y una pérdida progresiva en sus capacidades motrices.

El promedio de vida aproximadamente de una persona con esta enfermedad es de 3 a 8 años y en la actualidad aún no cuenta con cura, se desconocen las causas que la originan, sin embargo, se atribuye que el 95% son casos esporádicos y un 5% son cuestiones hereditarias. Según el grupo social Apoyo Integral Gila AC se estima que una de cada 20,000 personas en el mundo es diagnosticada con ELA y se calcula que en México hay 6,000 casos [1], [2] .

La sintomatología se puede presentar de maneras diferentes dependiendo el tipo de ELA, bulbar o medular-espinal, el tipo bulbar provoca problemas en la voz, la articulación de las palabras y la posibilidad de alimentarse de forma autónoma, ya que altera los músculos que empujan la comida a través del tubo digestivo, mientras que la medular-espinal se manifiesta con debilidad de las extremidades, caídas repentinas e imposibilidad de caminar; en ambos casos con el progreso de la enfermedad, las personas con ELA suelen presentar dificultades en el proceso de alimentación y respiración.

Las personas con ELA requieren de un plan de asistencia personalizado, conformado por equipos y terapeutas, según la asociación Familiares y Amigos de Enfermos de la Neurona Motora A.C. (FYADENMAC), el costo de un paciente con ELA se acerca a los 440,000 pesos mexicanos, cantidad que los servicios públicos de salud en México no cubren [3].

Entre los diferentes tipos de terapias, se encuentra la terapia ocupacional (TO), la TO juega un papel importante en el aumento de calidad de vida de las personas con ELA, ya que son un conjunto de técnicas que estimulan la independencia funcional [4].

La terapia ocupacional provee a las personas la posibilidad de interactuar de manera independiente en el hogar, un aspecto importante a considerar son las barreras arquitectónicas, ya que limitan el acceso a los dispositivos, como el control del clima, luces, apertura de puertas, cortinas, entre otras cosas indispensables para llevar a cabo las tareas domésticas [5].

En la actualidad existen tecnologías como sintetizadores de voz con seguimiento de mirada, que facilitan el proceso de comunicación. Sin embargo, surge la necesidad de una herramienta que asista a las personas con ELA en el control de los dispositivos del hogar [6].

Esta investigación estará enfocada en el apoyo a personas con ELA, se pretende mejorar la calidad de vida durante el proceso de la enfermedad, enfocándose en el diseño de un sistema que a través de la implementación de una aplicación para tableta y un periférico de seguimiento de mirada, permita a las personas interactuar con los dispositivos conectados a la red del hogar, así como la comunicación mediante mensajería por servicios web.

## 1.1 Antecedentes

ELA afecta las áreas de la médula espinal donde se localizan partes de las células nerviosas que envían señales para llevar a cabo el movimiento de los músculos, a medida que estas áreas se ven afectadas, se produce endurecimiento en los músculos lo que se denomina como (“esclerosis”).

La sintomatología que una persona con ELA puede presentar es la siguiente:

- Debilidad muscular, lo que origina discapacidad en el uso de manos, brazos, piernas o los músculos del habla.
- Tics y calambres musculares, especialmente en manos y pies.
- En etapas más avanzadas, falta de aliento y dificultad para respirar y tragar.

Existen algunas aplicaciones para personas con discapacidades motrices, como el sensor Tobii, el cual permite el control de la computadora con la vista y facilita la comunicación a través de los periféricos de la computadora [7].

Otra herramienta de gran utilidad es el tablero de comunicación de Tlacuany, que es un programa para la comunicación, que tiene un panel de escritura con un sintetizador de voz donde se puede escribir y al oprimir el icono del micrófono reproduce el texto con voz femenina o masculina [8].

En la actualidad no se cuenta con herramientas especiales que una persona con ELA necesita para desenvolverse independientemente en su día a día, como el acceso a los dispositivos básicos del hogar, encender las luces, controlar el clima, abrir y cerrar las cortinas, entre otros.

Como se puede observar existen diferentes plataformas para el apoyo en el desarrollo de calidad de vida de personas con ELA, sin embargo, hay una necesidad de un sistema a la medida con la capacidad de dar autonomía de acción sobre las cosas del hogar.

El sistema propuesto incluye la utilización de un sensor de seguimiento ocular (SO) o de mirada, que permita mediante una interfaz gráfica en una tableta, el autodomínio de las cosas del hogar, una casa inteligente capaz de ser operada a través del seguimiento de la mirada.

## **1.2 Planteamiento del problema**

ELA afecta el estado emocional de las personas, los problemas musculares son progresivos, por ser una enfermedad crónica ante una recaída no hay recuperación y su vida cotidiana se va tornando más dependiente de un cuidador.

El Internet de las cosas, y los sistemas embebidos abren un sin fin de posibilidades. La domótica y el desarrollo de sistemas basados en la filosofía incluyente de personas con necesidades especiales deben ser valorados como filosofía de las nuevas tendencias tecnológicas 4.0. "El control de un hogar inteligente a través del seguimiento de la mirada", sería un gran beneficio como herramienta de apoyo en la vida cotidiana de estas personas.

Por lo que surge la necesidad de tener una herramienta que apoye a las personas con ELA, así como a las personas que se encargan de su atención.

Lo cual nos lleva a la siguiente pregunta de investigación:

¿Cuáles son los elementos de un sistema para un ambiente autónomo de domótica, que sea fácil de controlar para las personas con ELA, mediante un sensor de seguimiento de mirada?

### **1.2.1 Objetivos (generales y específicos)**

Para lograr el desarrollo del sistema que propone este trabajo se han planteado los siguientes objetivos.

### **1.2.2 Objetivo Específico**

- Estudiar las diferentes temáticas que inciden en el estado del arte.
- Conocer el tipo de sensor a utilizar, para detectar el movimiento ocular.
- Seleccionar las cosas o dispositivos que se controlarán en el hogar.
- Determinar el tipo de dispositivo y tipo de comunicación entre el sistema central y las cosas del hogar.
- Analizar los tipos de servicios web para el envío de mensajería.
- Diseñar la arquitectura del sistema.
- Implementar un prototipo del sistema.

## **1.3 Objetivo General**

Implementar un sistema para personas con ELA, que permita el control domótico mediante la vinculación de un dispositivo de seguimiento de mirada y una tableta.

## **1.4 Justificación de la tesis**

En la rehabilitación es fundamental la terapia ocupacional, que es una herramienta importante para el tratamiento terapéutico de algunas discapacidades físicas, el desarrollo de un sistema supliría los déficits de invalidez devolviendo la capacidad de realizar tareas a las personas con ELA.

El objetivo es mejorar el bienestar de los pacientes, la mejora psicológica mediante el fomento de integración social, laboral y familiar, lo cual empodera a las personas con esta enfermedad, mejorando su salud mental y evitando cuadros de depresión emocional que sólo empeorarían la esperanza de vida, en la actualidad es deseable un sistema que apoye a las personas con ELA a desenvolverse en su medio.

## **1.5 Alcances y delimitaciones**

El sistema desarrollado considera el caso de una persona con ELA, con problemas de movilidad en sus extremidades, y sus movimientos oculares se encuentran en buen estado.

## **1.6 Metodología**

Fase 1: Investigación del estado del arte

- La primera etapa se basa en la revisión de fuentes bibliográficas, tomando en primera instancia el estudio de los sistemas basados en casas inteligentes para personas con ELA, considerando específicamente como campo de estudio personas limitadas a problemas motrices, “que se les facilite el manipular una tableta con la vista”.

Fase 2: Análisis y Diseño del sistema

- Utilizando las herramientas de ingeniería de software, se documenta mediante diagramas del modelo orientado a objetos el camino para cumplir con el objetivo, delimitando la cobertura del mismo, dejando claro a su vez los requerimientos funcionales, no funcionales y los componentes o herramientas de integración para completar la arquitectura del sistema.

Fase 3: Implementación y Pruebas.

- Se desarrolló la integración de los componentes, mediante un protocolo de pruebas para validar la funcionalidad del sistema, se analizaron los resultados con el objetivo de refinar el producto final, por último, se validaron los objetivos esperados para asegurar si la aplicación tuvo éxito o no.

## **1.7 Organización de la tesis**

El capítulo II de esta tesis presenta un estudio sobre los temas que abarca el sistema propuesto, los cuales son; el ELA, los sensores de movimientos oculares, domótica y los trabajos relacionados.

El capítulo III presenta la arquitectura propuesta, el análisis y diseño del sistema de SO, que incluye los requerimientos de información y el modelo lógico de interacción con las cosas del hogar.

El capítulo IV presenta una descripción de implementación del sistema con los pacientes de ELA, el cual consiste en la utilización del sistema, la interacción con las cosas del hogar que contemplará el sistema de domótica.

El capítulo V presenta el análisis de resultados al que se llegó después de haber implementado el sistema con un paciente de ELA.

Finalmente, el capítulo VI presenta las conclusiones obtenidas de la presente investigación y el trabajo futuro de la tesis.

# 2 Estado del arte

## 2.1 Introducción

ELA (esclerosis lateral amiotrófica) es un tipo de enfermedad que afecta las neuronas motoras, incapacita a las personas para llevar a cabo el movimiento de los músculos del cuerpo y en algunos casos severos, afecta la respiración y deglución de alimentos, sin embargo, el sentido de la vista, gusto, tacto y olfato no suelen ser afectados.

Se busca implementar un sistema y una interfaz que permita interactuar al usuario con ELA con el sistema, por lo que se deben incluir diversos componentes; entre ellos un sensor que relacione la vista con los iconos del programa, con lo cual se le permitirá al usuario desencadenar eventos como si fuesen un clic del ratón. Las coordenadas de la mirada estarán relacionadas con los iconos o los módulos a los cuales accederá el usuario. Asimismo, la comunicación entre el sistema central y las cosas que se van a controlar será a través de diferentes módulos de comunicación.

De tal manera que en este capítulo se indagará en temas referentes a la propuesta de tesis como son: ELA, domótica, el internet de las cosas (IOT), sistemas embebidos, dispositivos de seguimiento de mirada, servicios web y trabajos relacionados.

## 2.2 ELA (esclerosis lateral amiotrófica)

ELA es una enfermedad que afecta las motoneuronas, estas son un tipo de células del sistema nervioso que se encuentran en el cerebro, se encargan de producir estímulos en diferentes músculos del organismo, es una enfermedad poco frecuente, progresiva y a la larga es mortal, cuando estas neuronas enferman y mueren conducen a la parálisis de movilidad.

En las primeras etapas la enfermedad se manifiesta con caídas repentinas, mientras el debilitamiento de las piernas va avanzando se van dando una serie de cuadros clínicos hasta imposibilitar el caminar, el deglutir alimentos, incluso el dejar de hablar y respirar en casos graves, la secuencia



con la que se presentan los síntomas de la enfermedad puede variar en diferentes como cuadros sintomáticos.

- **Común:** suele iniciarse de forma asimétrica por una extremidad superior.
- **Pseudopolineurítica:** se inicia por las extremidades inferiores y va progresando de forma ascendente lentamente.
- **Bulbar:** evolución más rápida y prácticamente desde el principio hay disfasia (dificultades en el habla), disfagia (dificultades para tragar), salivación excesiva y aumento de las mucosidades.

ELA es una enfermedad de las células nerviosas (neuronas), que se encargan de controlar el movimiento de los músculos, la enfermedad es progresiva e imposibilita movimientos como masticar caminar, hablar y otras actividades o capacidades motrices, los síntomas son progresivos por lo que suelen empeorar a través del tiempo.

Las neuronas motoras o motoneuronas son células nerviosas que se extienden desde el cerebro hacia la medula espinal y los músculos, La degeneración de las neuronas motoras repercute directamente en la movilidad muscular, inicialmente comienzan con debilitamiento como la incapacidad de realizar el movimiento como el abotonar una camisa.

La capacidad de controlar voluntariamente los músculos se va viendo afectada. El hablar comer o respirar se va dificultando, el promedio de vida una vez que es diagnosticada una persona con ELA es de 3 a 8 años, el ELA afecta a todas las razas, sin embargo, presenta más incidencia en mujeres y en personas de raza blanca.

En la mayoría de los casos se presenta de forma esporádica, según el Instituto Nacional de Trastornos Neurológicos y Accidentes Cerebrovasculares del 5 al 10% de los casos se atribuye a casos genéticos, "familiares".

Los síntomas de la enfermedad pueden pasar desapercibidos, al principio contracciones musculares involuntarias, debilidad muscular, caídas repentinas, etc. Como se ha venido documentando la secuencia de los síntomas pueden ser diversos; problemas en movilidad de brazos o piernas, al hablar, hasta el imposibilitar la independencia al deglutir y respirar.

La electromiografía (EMG) es un método invasivo para monitorear la

actividad muscular mediante una aguja o electrodo, que es usualmente utilizado para apoyo en el diagnóstico. La resonancia magnética (RM), es otro tipo de método no invasivo para obtener información del cerebro y medula espinal en el caso del ELA.

Las causas del ELA se desconocen, el padecimiento es aleatorio ya que no se sabe en realidad por que se da en algunas personas y otras no, hay estudios genéticos y de factores ambientales que no denotan un gran impacto en la atribución de la enfermedad.

ELA no tiene cura, pero la intervención de diferentes terapeutas con un programa a la medida del paciente logra la mejora significativa en la calidad de vida de estas personas, el darles la posibilidad de ser independientes en su medio es un impacto vital en su salud.

Un programa complementado por medicamentos, terapia física, plan nutricional, apoyo de respiración, son esenciales como tratamiento de la enfermedad, sin embargo, son costosos y van en aumento en relación a las necesidades que presentan los pacientes con el progreso de la enfermedad. En la actualidad hay investigaciones genéticas que son prometedoras para frenar el avance de la enfermedad, sin embargo, es impredecible si funcionara [9].

## 2.3 Domótica

El internet de las cosas (IOT) es un nombre para dispositivos conectados en red, en el hogar o cualquier lugar de trabajo. En el hogar contamos con computadoras embebidas en diferentes dispositivos, lavadoras, refrigeración o clima, refrigerador, microondas, televisión, sin embargo, solo algunas de ellas pueden llegar a tener (IOT), esto es, todo aquel dispositivo que se encuentre de alguna forma conectada a la red.

Las principales características que definen el (IOT) son:

- **Redes:** que los dispositivos se comuniquen entre sí, o a servidores ubicados en la red local o en internet.
- **Detección:** detección mediante sensores para percibir eventos en el entorno.

- **Actuadores:** Dispositivos que hacen algo. Apertura o bloqueo de puertas, encender el televisor o aires, control de iluminación, entre otros.

La domótica es la automatización del hogar, que es aumentar el confort mediante la reducción de la interacción del humano en procesos del hogar, actividades como regular el clima, controlar el televisor, aumentar y disminuir la intensidad de la iluminación, a través de la incorporación de sistemas embebidos en las cosas del hogar [10].

Hoy en día existe una gran variedad de dispositivos disponibles para la automatización del hogar. Se pueden adquirir un kit de automatización en tiendas físicas o en línea (online) y de fácil instalación, controlables a través de asistentes virtuales, como sets de focos o contactos, sin embargo, muchos son caros e incluso imposibles de personalizar para las propias necesidades del usuario final.

Arduino es una plataforma para construir prototipo sistemas embebidos, ideal para sistemas de domótica, facilita la integración de funcionalidad a las cosas del hogar, mediante el uso de sensores, actuadores y programación, permite el despliegue a la medida sin altos conocimientos técnicos que usualmente son escalados a muchos sistemas de automatización en el hogar [11].

## 2.4 Sistemas Embebidos

Los sistemas embebidos son esenciales en la domótica ya que forman parte del internet de las cosas, básicamente es una placa de circuito impreso (PCB), adicionado con sensores, actuadores, módulos de comunicación y software, dedicados a llevar a cabo tareas específicas, como la apertura de la llave del gas al detectar una fuga, el cierre de cortinas, etc. [12].

A continuación, en la Ilustración 2.1 Sistema embebido se describe la función de un sistema embebido en una bombilla, cuenta con un sensor acústico que al percibir un aplauso a través del sensor la lógica programada manda a encender la bombilla por medio de un actuador, como un relevador. Este sistema embebido en la bombilla a su vez forma parte de los nodos de la red del hogar inteligente y su comunicación puede ser configurada por diferentes tecnologías de comunicación alámbrica, inalámbrica o híbrida.

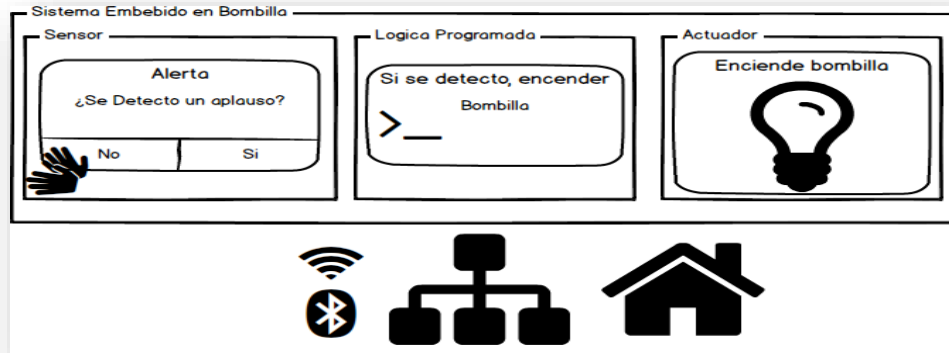


Ilustración 2.1 Sistema embebido

## 2.5 Sensores

Un sensor es un dispositivo adecuado para detectar variantes en magnitudes químicas o físicas y transformarlas consecuentemente en variables eléctricas en relación con la cantidad que se está midiendo, por ejemplo, un elemento para medir temperatura, la capacidad para transformar la entrada de temperatura en un cambio de resistencia para determinar el calor térmico en una métrica determinada [13].

Los sistemas de monitorización y control requieren sensores para medir cantidades físicas tales como posición distancia, fuerza, deformación, temperatura, vibración, aceleración, etc. [14].

## 2.6 Tipo De Sensores

Existen diferentes tipos de sensores, con funcionalidades diferentes, todo depende de la variable que se requiere medir en relación al propósito u objetivo del proyecto.

- **Sensor de temperatura:** la temperatura puede ser medida detectando algún cambio en una propiedad física, estos son utilizados en diversas aplicaciones como la elaboración de alimentos, climatización, dispositivos médicos, se utilizan para asegurar el cumplimiento de un proceso en un rango deseable.
- **Sensor de movimiento:** el sensor de movimiento detecta patrones en determinadas áreas, la implementación se puede dar en algunos

sistemas de alarmas de seguridad doméstica.

- **Sensor de distancia:** este detecta la distancia desde el punto de un objeto hasta su objetivo, un ejemplo es el LiDAR (Light Detection and Ranging), utilizado en algunos procesos de seguridad para mantener un umbral de seguridad.
- **Sensor de imagen:** es el elemento de una cámara electrónica, tanto de video como de fotografía estática, que detecta y captura la información que compone la imagen.

En la Ilustración 2.2 se pueden observar diferentes tipos de sensores y su aspecto físico.



Ilustración 2.2 Sensores, aspecto físico.

## 2.7 Actuadores

Son dispositivos capaces de transformar energía hidráulica, neumática o eléctrica en la activación de un proceso con la finalidad de generar un efecto sobre un elemento externo. Recibe una orden y en función a ella activa un elemento de control como una válvula.

Existen varios tipos de actuadores, los hay electrónicos, hidráulicos, neumáticos, motores, etc., como se muestra en la ilustración 2.3.

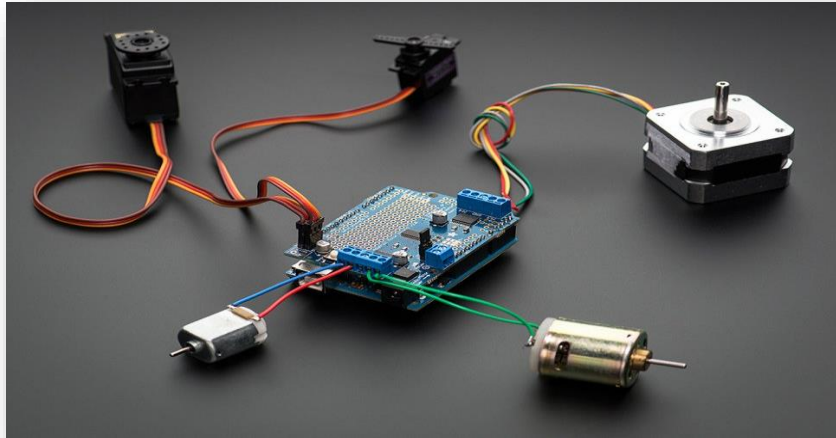


Ilustración 2.3 Tipos de actuadores

## 2.8 Comunicación embebida

La conectividad entre los dispositivos es la característica principal de gestión sobre los nodos que se encuentran dentro de una red, la conexión entre estos puede ser física, inalámbrica o híbrida, la comunicación forma parte del sistema embebido y es lo que diferencia entre un dispositivo normal y un dispositivo IOT [21].

Las comunicaciones es un aspecto importante para los proyectos (IOT), hay una balanza entre diferentes aspectos en compensaciones que se deben tener en cuenta dependiendo el protocolo de comunicación a utilizar, a mayor velocidad de datos, necesita mayores recursos de procesador y por lo tanto es mayor el consumo eléctrico, los dispositivos IOT que tienen soporte TCP/IP se encuentran categorizados con mayor consumo de energía que conexiones de tipo Bluetooth y Zigbee.

Dentro de las topologías de redes como se ha venido mencionado existen diferentes tipos de comunicación, en ocasiones por cable sin embargo la gran mayoría de las veces es difícil lograr este tipo de redes debido a la ubicación de los nodos de la red, el entorno hace que el cableado no sea práctico, en su gran mayoría las redes requieren de prácticas con tecnología inalámbrica en la transmisión de datos por medio de Radio Frecuencia (RF) en distintos protocolos de comunicación.

Los tipos de redes inalámbricas pueden variar según su arquitectura:

- **Redes de malla:** utilizados para enviar datos de un nodo a otro mediante puntos intermedios.
- **Punto a Multipunto:** este tipo de comunicación es de punto a multipunto, uno a uno o de muchos a muchos.
- **Redes Celulares:** está compuesta por red celular en base a una estación de un operador de telefonía cuya cobertura abarque grandes áreas geográficas.

Los nodos de sensores pueden clasificarse a su vez en diferentes roles según su uso dentro de la red.

- **Nodos de sensor básico:** tiene un solo sensor y mecanismo de comunicación solo se encargan de pasar los datos a otro nodo en la red, se clasifican como las hojas dentro de la red de nodos en la red.
- **Nodo de datos:** son nodos sensores que almacenan datos, pueden enviar o no los datos a otro nodo para posteriormente guardarse en alguna base de datos o darles salida en una interfaz, panel o medidor LED, la esencia de este tipo de nodos es el registrar o representar los datos.
- **Nodo Coordinador:** este nodo es responsable de la administración de direccionamiento, e intercambio de información entre los demás nodos, es el portal de inicio en el escaneo de la red, todo depende de la topología de red y delegación de responsabilidades de los nodos dentro del diseño de la topología de la red.

## 2.9 Módulos de radio frecuencia

Los módulos de RF son transmisores y receptores inalámbricos que podemos emplear como forma de comunicación entre los nodos de nuestra red para procesar información, el alcance depende del voltaje de alimentación y la antena que empleemos en estos.

- **ESP8266** es un chip integrado con conexión Wi-Fi, compatible con el protocolo TCP/IP.
- **ESP32** es un módulo de comunicación con la tecnología de bluetooth y WIFI.
- **XBee** es un módulo con múltiples protocolos inalámbricos y frecuencias de RF, incluidos estándares abiertos de celulares, Zigbee, Thread, 802.15.4, WiFi.

En la ilustración 2.4 se pueden ver las características de los módulos anteriormente mencionados.



Ilustración 2.4 Módulos de comunicación

Algunos módulos se pueden adquirir en placas con relevadores integrados, los cuales se pueden programar para ejecutar ciertas funcionalidades y además cuentan con un relevador integrado para controlar el flujo de corriente en el dispositivo IOT.

## 2.10 Módulo ESP8266

ESP8266 es un chip microcontrolador Wi-Fi de bajo costo que tiene la capacidad de potenciar IOT y ayuda al intercambio de información entre varios objetos conectados, en la Ilustración 2.5 se muestra este chip. Es compatible con el protocolo TCP/IP, es de los módulos más baratos en el



mercado y una gran ventaja es el bajo consumo de corriente eléctrica [15].

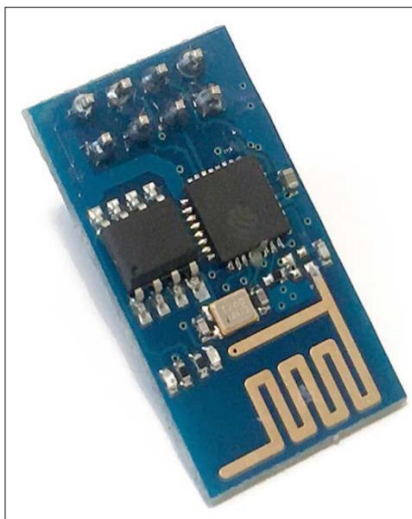


Ilustración 2.5 ESP8266

Al inicio de su comercialización por ser de origen chino, de la compañía Espressif Systems, con sede en Shanghai, era muy complicado adoptarlo, debido al desafío que requería el comprender el idioma para poder comprender su funcionalidad, sin embargo en la actualidad es fácil encontrar documentación en inglés y español, el módulo se puede programar con el IDE de Arduino y también existe la posibilidad por medio de comandos AT, este módulo se puede encontrar en circuitos integrados con relevadores con la finalidad de una rápida aplicación en proyectos de prueba.

## 2.11 Módulo ESP32

El módulo ESP32 es un chip microcontrolador de comunicación híbrida ya que puede interactuar con otros sistemas tanto por Wi-Fi como con bluetooth, este tipo de módulo tiene un diseño robusto ya que puede funcionar correctamente en temperaturas que van desde  $-40^{\circ}\text{C}$  a los  $125^{\circ}\text{C}$ .

Existen diferentes modelos, algunos para programarse para utilización en pruebas y versiones minimalistas para integración en productos PCB a la medida, cuentan con una antena y amplificador integrada, y una filosofía de consumo de energía baja.

Existen diferentes tipos de módulos ESP32, uno de ellos se observa en la Ilustración 2.6, este tipo de módulos traen un puerto USB, con la finalidad de facilitar la carga del código o programación del mismo, también cuenta con

botones que al realizar una serie de pasos entra en modo programación, sin necesidad de componentes extras conectados a un protoboard.



Ilustración 2.6 ESP32

Cuenta con gran variedad de recursos; lenguajes de programación, frameworks, librerías, código/ejemplos, entre otros. Los más comunes a elegir son: Esp-idf (Espressif IoT Development Framework) desarrollado por el fabricante del chip, Arduino IDE (en lenguaje C++), Simba Embedded Programming Platform (en lenguaje Python), RTOS's (como Zephyr Project, Mongoose OS, NuttX RTOS), MicroPython, LUA, Javascript (Espruino, Duktape, Mongoose JS), Basic [16].

## 2.12 Módulo XBEE

Un XBee es un módulo de comunicación que utiliza radiofrecuencia (RF) para intercambiar datos entre módulos XBee. Los módulos XBee transmiten a 2,4 GHz o 900 MHz de largo alcance y tienen su propia red de protocolos. El módulo XBee en sí es muy pequeño, lo que proporciona una fácil adecuación como parte de los nodos embebidos en la red.

Los módulos también son de baja potencia y pueden usar un modo de suspensión especial para reducir aún más el consumo de energía. El módulo XBee puede monitorear sus pines de datos y transmite los datos leídos a otro módulo XBee.

El rango de alcance de línea de vista oscila entre los 1200 metros y

3200, se pueden realizar diferentes tipos de topología de red dependiendo de la compatibilidad de los protocolos de la familia del módulo, en la actualidad son configurables desde lenguajes programación de alto nivel como lo es Python, lo que reduce la curva de aprendizaje para comprender su funcionamiento.

Existe una gran variedad de módulos de la familia Xbee, algunos de tipo celular que funcionan mediante un chip de telefonía, también se pueden adquirir kits de aprendizaje y desarrollo, algunos con antena externa para mayor alcance, otros son de tamaño mínimo para ser soldados en placas PCB como lo es la generación Xbee 3 tal y como se muestra en Ilustración 2.7.

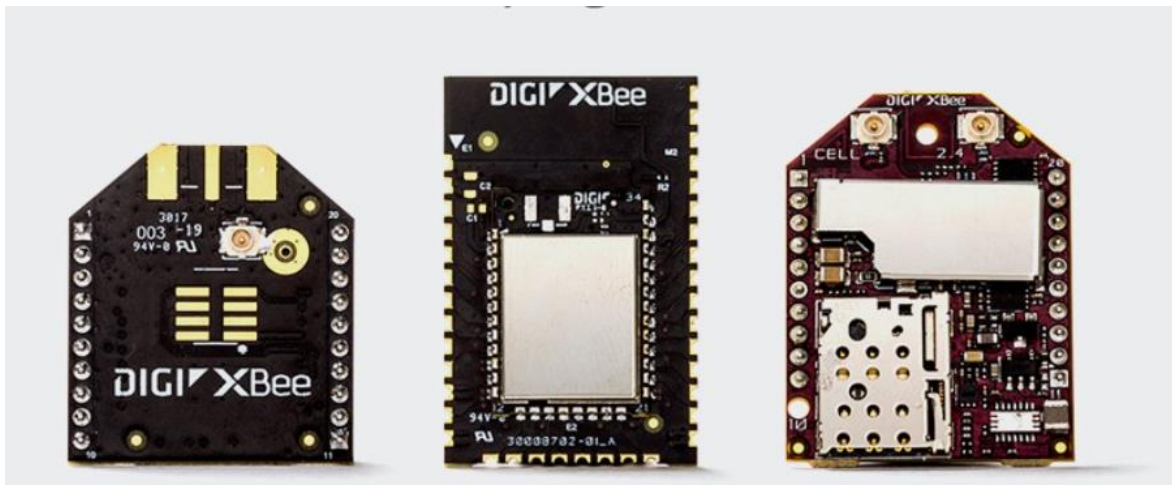


Ilustración 2.7 Módulos xbee

La configuración inicial se hace desde Digi XCTU que es una interfaz gratuita y sencilla para configurar módulos Digi XBee, aquí se asigna una dirección, se establecen los valores predeterminados del tipo de perfil de comportamiento que tendrá en la red, se pueden cargar configuraciones y exportar ajustes previamente configurados, etc.

## 2.13 Módulos Comunicación Celular

Los módulos celulares son diseñados para incluirse en proyectos de todo tipo, este tipo de dispositivos utilizan tecnología satelital por lo cual necesitan de un chip inmerso dentro de ellos como se puede apreciar en la

## Ilustración 2.8.

Los módulos de comunicación celular son excelentes para áreas remotas donde no es tan fácil el acceso a las tecnologías, como lo pueden ser construcción de presas, campamentos en minas, flotillas de carros, unidades no fijas, también en el sector acuícola son incorporadas en ocasiones a sistemas para medir la temperatura del agua, también en algunas aplicaciones en el monitoreo de alimentos en la agricultura.

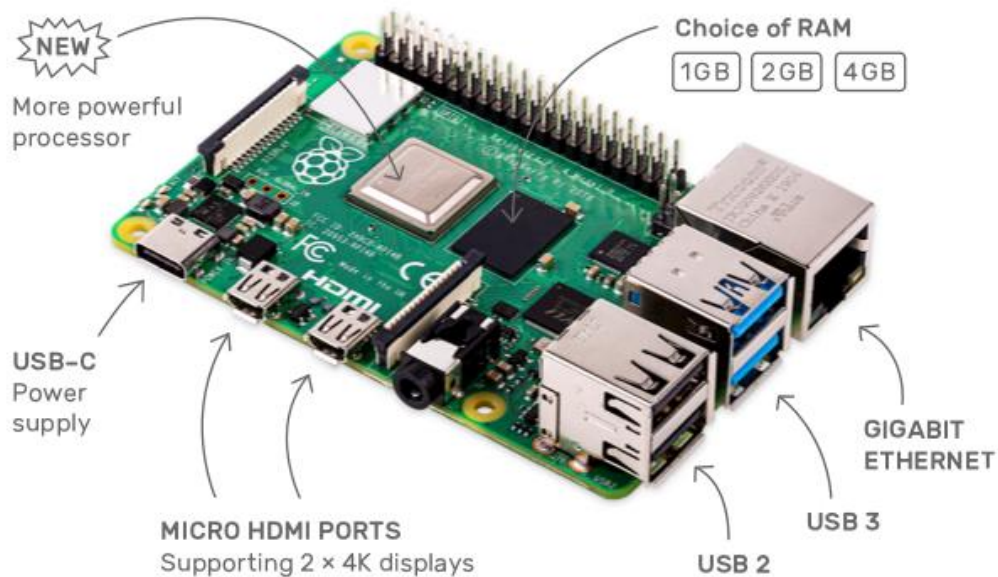
Este tipo de módulos se encuentran certificados por lo cual el tiempo de integración es rápido, algunos modelos traen diferentes tipos de comunicación como lo es bluetooth. Xbee tiene una línea dedicada a este tipo de comunicaciones, como lo es el modelo XB-CA2-UT001 que cuenta con compatibilidad blueetooth y diferentes proveedores de comunicación satelital como AT&T/Verizon/Bell/Telus, es importante conocer el tipo de operador con soporte en la zona a implementar.



Ilustración 2.8 XBEE Celular Modelo XB3-C-A2-UT-001

## 2.14 Coordinador domótico

Un coordinador de los dispositivos inteligentes en el hogar suele ser una computadora de bajo costo, que es una placa base con los componentes necesarios para comportarse como una computadora común, pero de tamaño reducido, centrándose básicamente en encapsular una gran capacidad de cómputo dejando por un lado accesorios que afecten sus dimensiones, como lectoras de CD, DVD, discos duros y todo lo que un computador personal requiere, un ejemplo de ello es Raspberry como se muestra en la Ilustración 2.9.



**Ilustración 2.9 Raspberry-pi 4**

Existen diferentes sistemas operativos para esta computadora, Raspberry en la actualidad tiene de manera nativa Raspbian, sin embargo, debido al auge de los sistemas embebidos y el internet de las cosas se han lanzado nuevos sistemas operativos, como lo es el caso de Windows con el sistema operativo Windows IOT.

El cual es compatible con la arquitectura de ARM, este tipo de arquitectura se identifican por su bajo calentamiento, en su mayoría libres de ventiladores de fábrica y bajo consumo energético.

Windows IOT es liberado con soporte en desarrollo de experiencias de aplicaciones Universal Windows Platform (UWP) como se muestra en la Ilustración 2.10, son aplicaciones de fácil adaptación de interfaz de usuario responsivo a diferentes tamaños de pantalla, programable en C#, C++, Visual Basic, javascript y en la interfaz de usuario utiliza XAML, Html o DirectX, además puede acceder a su API universal para ejecutar su aplicación desde cualquier dispositivo independientemente si su desarrollo fue nativo en Windows 10.

El IDE de Visual Studio provee la capacidad de desarrollar aplicaciones UWP e instalarlas desde su entorno directamente en el sistema embebido destino, solo con apuntar a la dirección IP del dispositivo en la red y acceder sus credenciales.



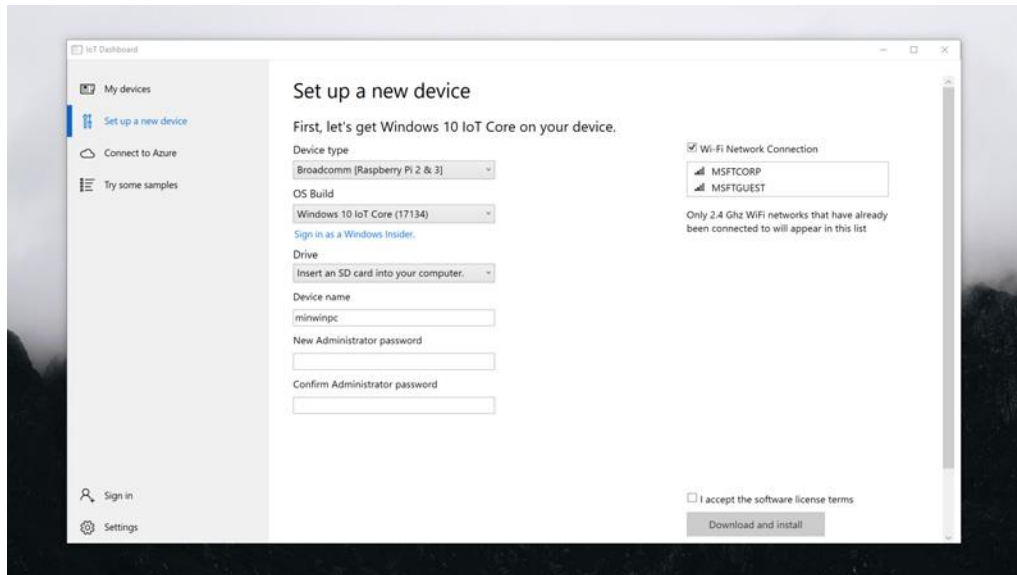
Ilustración 2.10 UWP multiplataforma

La instalación de Windows IOT se realiza a través de Windows IOT Dashboard, una interfaz se muestra en la Ilustración 2.11, la instalación del sistema operativo se realiza en una memoria micro SD, la rapidez del funcionamiento del sistema operativo va en relación a los recursos de velocidad de escritura y lectura de la memoria.

Desde el mismo dashboard se puede realizar la instalación de aplicaciones o dar mantenimiento de software, sin necesidad de acceder físicamente al minicomputador, actualmente la compatibilidad del sistema operativo se encuentra limitado a los siguientes computadores.

Los dispositivos admitidos son:

- Intel [MinnowBoard Turbo/MAX (x64)]
- Broadcomm [Raspberry Pi 2 & 3]
- Qualcomm [DragonBoard 410c]
- NXP [i.MX6/i.MX7/i.MX8]



**Ilustración 2.11 Windows IOT dashboard**

Por otra parte, es posible correr Windows 10 en Raspberry pi 3, proyecto creado por entusiastas de las tecnologías ajenos a Microsoft Corporation, que ponen a la disposición de la red versiones basadas en la arquitectura ARM, sin embargo, el rendimiento es demasiado lento y la fuente no es oficial.

Se espera próximamente sea posible poder instalar tanto Windows IOT como la versión de Windows 10 de entusiastas en la nueva Raspberry 4, para poder obtener el máximo provecho de rendimiento. En la Ilustración 2.12 se puede apreciar las versiones disponibles y el ecosistema de instalación [17].

UUP volcado v3.33.5

Casa Descargas Inglés Estados Unidos Modo oscuro Código fuente Discordia

## Volcado UUP

Descargue archivos UUP de los servidores de Windows Update con facilidad.

Search for builds...

Opciones avanzadas

Explore la lista de compilaciones conocidas  
Elija una compilación que ya se conoce en la base de datos local y descárguela.

Obtener la última compilación  
Recupere la última información de compilación de los servidores de Windows Update.

Construcciones recientemente agregadas

Construir	Arquitectura	Fecha Agregada
Windows 10 Insider Preview 19025.1 (vb_release) arm64	arm64	2019-11-15 18:30:23 UTC
Windows 10 Insider Preview 19025.1 (vb_release) x86	x86	2019-11-15 18:19:39 UTC

Ilustración 2.12 Windows 10 raspberry

## 2.15 Prototipado de embebidos

Arduino, en la ilustración 2.13 se muestra su placa, es una plataforma creada con la finalidad de proporcionar un fácil entorno de desarrollo de prototipos de hardware y software de código abierto, es el microcontrolador ideal para crear proyectos sin necesidad de un gran conocimiento de electrónica, proporciona un entorno de desarrollo integrado (IDE) para escribir, compilar y cargar programas llamados sketch's en el microcontrolador a través de USB desde un computador, el lenguaje se llama Arduino y es similar a C, es ideal para prototipar todo tipo de proyectos, desde aficionados hasta creaciones que requieren mayor complejidad.



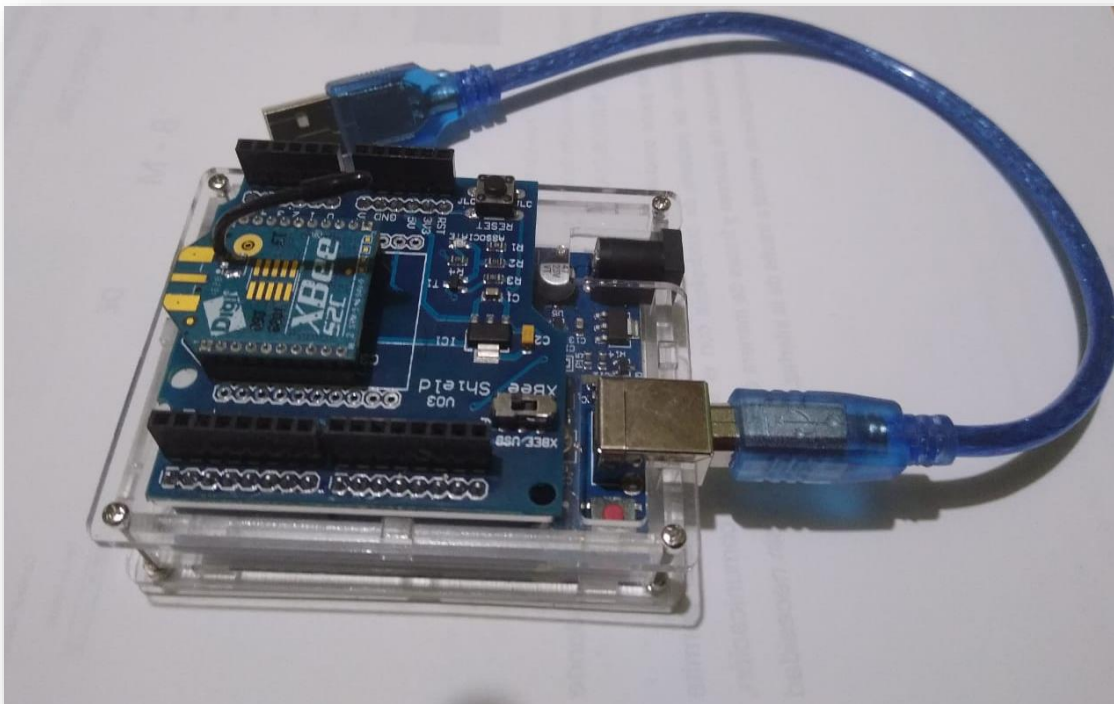
Ilustración 2.13 Arduino



Las ventajas es ahorrar tiempo de inversión y conocimiento en el diseño de un PCB, les dan la posibilidad a personas sin altos conocimientos en electrónica de generar pruebas pilotos, sin embargo, sus capacidades de hardware son limitadas a comparación de un Raspberry-pi, usualmente son utilizadas como nodo en tareas que no requieren gran procesamiento.

Arduino es ideal para realizar pruebas de módulo embebido, funcionalidades de nodos, subordinados de un coordinador de red, como puede ser un sensor de temperatura que a su vez con un módulo de comunicación y un termómetro este guardando el historial de temperatura en base de datos, etc.

Las adaptaciones de módulos de comunicación se pueden realizar a través de shield's, o hat's como se muestra en la Ilustración 2.14, son adaptadores que facilitan el intercalado de distintas tecnologías, en la ilustración 2.14 se puede apreciar un Arduino con un adaptador dedicado para la comunicación mediante el módulo Xbee.



**Ilustración 2.14 Adaptaciones Arduino**

## 2.16 Sensores de seguimiento de ojos

Quien padece ELA conserva los movimientos oculares durante el progreso de la enfermedad. El uso de las tecnologías brinda la posibilidad a las personas con ELA de poder interactuar de una manera virtual con su entorno físico.

En la actualidad existen dispositivos de seguimiento de mirada, destinados hacia diferentes objetivos como lo son el Neuromarketing, fines lúdicos como la inmersión en video juegos y el uso del mouse y teclado del computador a través del control ocular [18].

Los dispositivos de seguimiento de mirada lanzan rayos infrarrojos hacia los ojos, los cuales generan reflexiones que a su vez son captadas por las cámaras del dispositivo para determinar la locación de la mirada como se observa en la ilustración 2.15, mediante un algoritmo se calcula la ubicación en la pantalla donde se encuentra alojado el dispositivo [19].

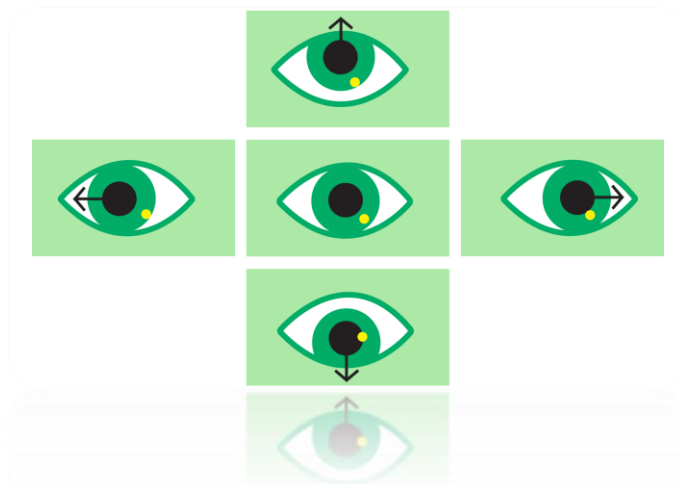


Ilustración 2.15 Reflexión Luz Infrarroja SO

IrisBound DUO, mostrado en la Ilustración 2.16, permite el control del computador con el movimiento de los ojos, compatible con una gran variedad de programas y aplicaciones a modo de herramientas de Comunicación Aumentativa y Alternativa (CAA) [20].



**Ilustración 2.16 IrisBound**

TM5 mini mostrado en la Ilustración 2.17, permite el seguimiento de ojos, robusto y fácil de usar, soportado por el sistema operativo de Windows [21].



**Ilustración 2.17 TM5 mini**

El rastreador ocular basado en pantalla SMI RED-m como se muestra en la ilustración 2.17, es un laboratorio ultraligero de SO totalmente portátil, alimentado por USB, de SensoMotoric Instruments (SMI). Está diseñado para investigadores que requieren la máxima movilidad y flexibilidades combinadas con datos confiables y el conjunto integral de características y métricas de un rastreador ocular de grado científico [22].

El Tobii Eye Tracker 4C el dispositivo mostrado en la ilustración 2.18 ofrece una experiencia inmersiva, es el único periférico para juegos de PC con seguimiento simultáneo de ojos y cabeza [23].



**Ilustración 2.18 Tobii eye tracker 4C**

## 2.17 Librerías Seguimiento Ocular

En la actualidad existen diferentes técnicas para el seguimiento ocular, algunas con fines de marketing dentro de una página de comercio electrónico, otras para registrar la experiencia de usuario en una página web, pero en la mayoría de los casos básicamente es detectar en que fija la atención el usuario.

Podemos encontrar algunas librerías de Javascript como WebGaze.js que es una biblioteca de seguimiento ocular que utiliza cámaras web como se puede observar en la Ilustración 2.19, su funcionalidad se basa en calcular las ubicaciones de los ojos de los visitantes web en una página en tiempo real. El modelo de seguimiento ocular que contiene se calibra automáticamente al observar a los visitantes de la web interactuar con la página web y entrena un mapeo entre las características del ojo y las posiciones en la pantalla [24].

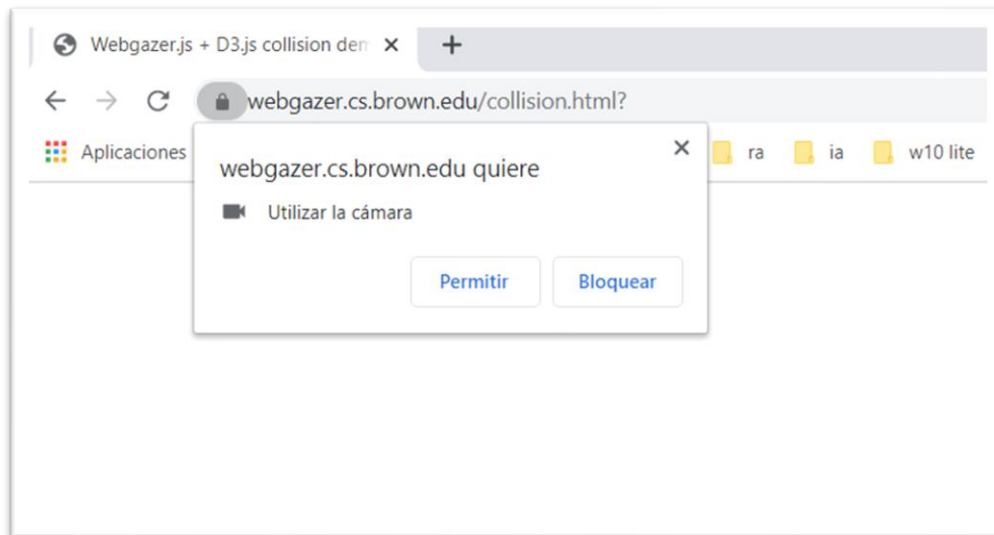


Ilustración 2.19 Permisos de periféricos

Esta tecnología es soportada por la mayoría de los navegadores como lo son Google Chrome, Microsoft Edge, Mozilla Firefox, Opera, Safari, en la página oficial vienen algunos ejemplos para descargar y probar, pero la exactitud de seguimiento depende de la cámara de cliente como se muestra en la Ilustración 2.20.

Este tipo de tecnologías ofrece una rápida integración con cualquier proyecto web personal, ya que en la página oficial puedes obtener la documentación y un tutorial de implementación explicando paso a paso la incorporación en los proyectos personales, así como el archivo fuente de webgazer.js.

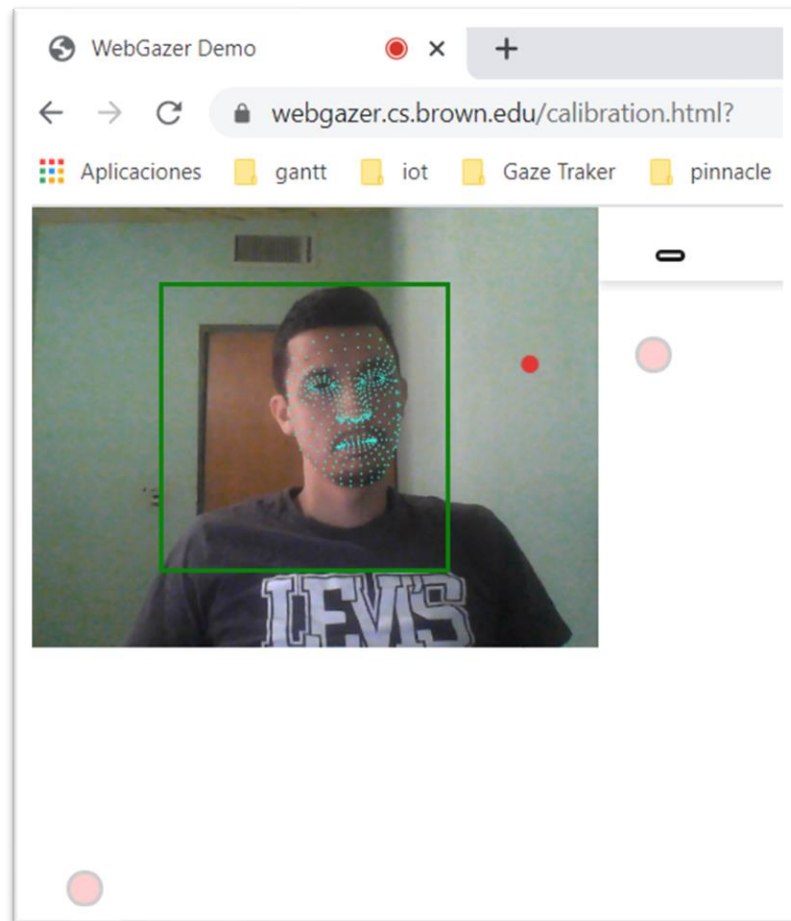


Ilustración 2.20 Calibración WebGaze

## 2.18 Servicios Web

Las Application Programming Interface (API'S) son interfaces que permiten la comunicación entre dispositivos manteniendo una capa de abstracción. Su objetivo es ser eficientes y simples de utilizar en los sistemas que las adoptan, suelen ser dedicadas para proporcionar escalabilidad en sistemas distribuidos, pueden ser consumidos e implementado en casi

cualquier tipo de tecnología.

Existen diferentes tipos de servicios Web, en algunas puedes consultar el clima, el precio actual del dólar, reconocer rostros a partir de una imagen o detectar emociones mediante el reconocimiento de imágenes, etc.

En la actualidad existen API'S dedicadas para facilitar el envío de mensajería instantánea como mensajes por WhatsApp, Facebook Messenger, y servicio de mensajería cortos (SMS).

Twilio es una de las plataformas de telefonía en la nube más potente que actualmente está disponible en el mercado, por otra parte se encuentra NiceApi.net, la cual ofrece un servicio gratuito de prueba, con la oportunidad de ampliar los beneficios [25], [26].

En la Ilustración 2.21, se puede observar un mensaje de recepción en WhatsApp desde un explorador configurado para poder usar la aplicación a través de la validación de lectura de un código QR como medida de seguridad, en este caso el envío de mensajería se realizó a través de la API de nice.api, un evento relacionado con el encendido y apagado de un foco.

Ofrece una versión gratuita para contactos limitados, sin embargo, al pagar puedes personalizar una gran agenda de contactos, nice.api cuenta con soporte en diferentes lenguajes de programación por lo cual se puede implementar en aplicaciones escritas en PHP, C #, ASP.NET, Java, Go, C++, etc.

Una ventaja de esto es que puedes enviar mensajes de forma autónoma por parte de los nodos ya que al tener librerías en C++ puede integrarse en la programación de módulos nodos en la red en placas computadoras de bajo costo, algunas aplicaciones prometedoras son: sensores de movimientos, detección de fugas de gas y poder notificar por diversos medios al propietario del hogar inteligente el evento desencadenado para así tomar precauciones.

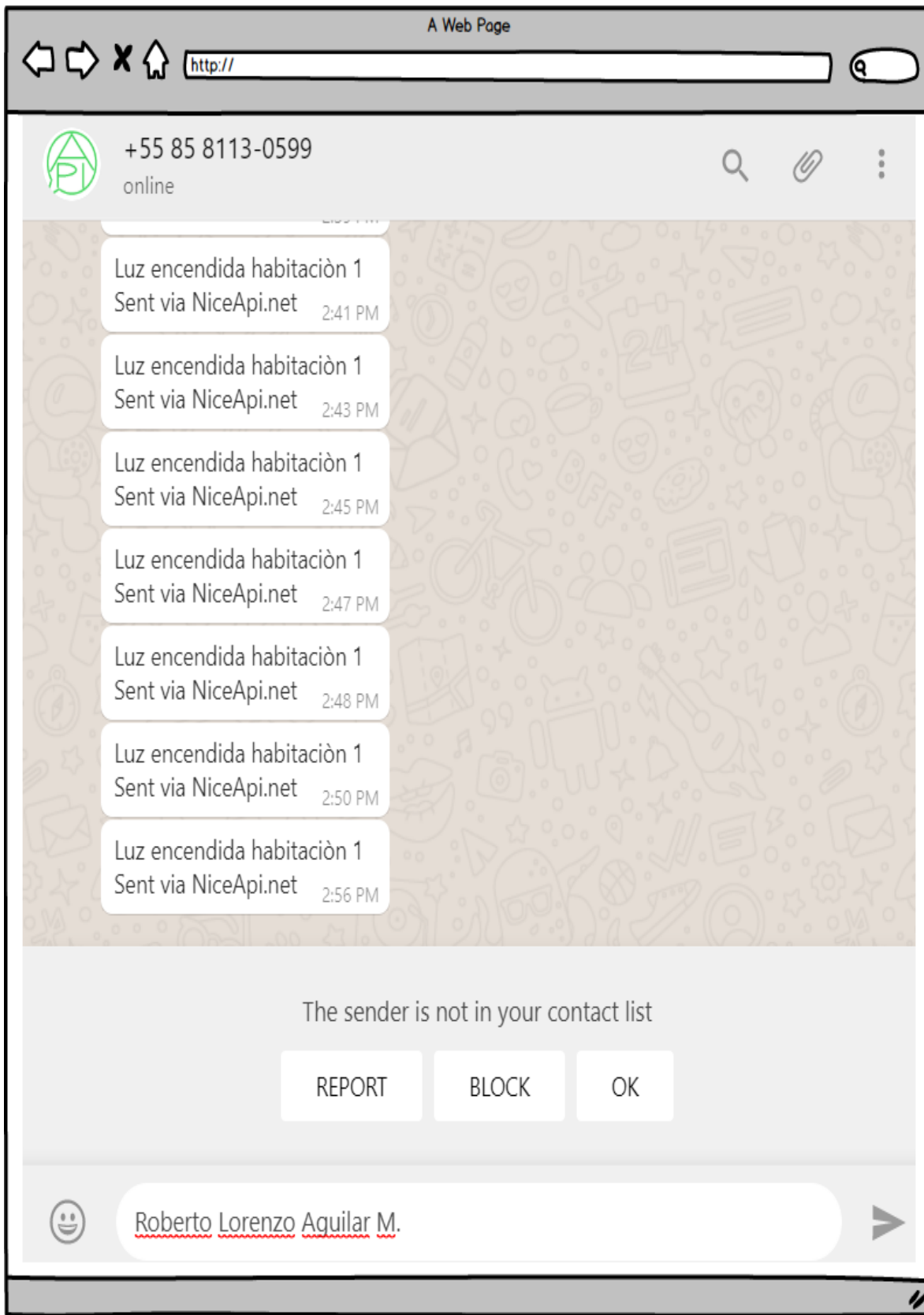


Ilustración 2.21Recepción Mensajería NiceApi.net

## 2.19 Trabajos relacionados

En la actualidad existe una gran variedad de dispositivos para aumentar el confort en el hogar, algunos paquetes iniciales para el hogar ( en inglés “home starter pack”) como se puede observar en la Ilustración 2.22, WYZE Smart Home Starter pack incluye cámara, sensor de movimientos, control de bombillas, enchufes, la conectividad es a través del sistema operativo de android y ios [27].



Ilustración 2.22 Kit Inteligente WYZE

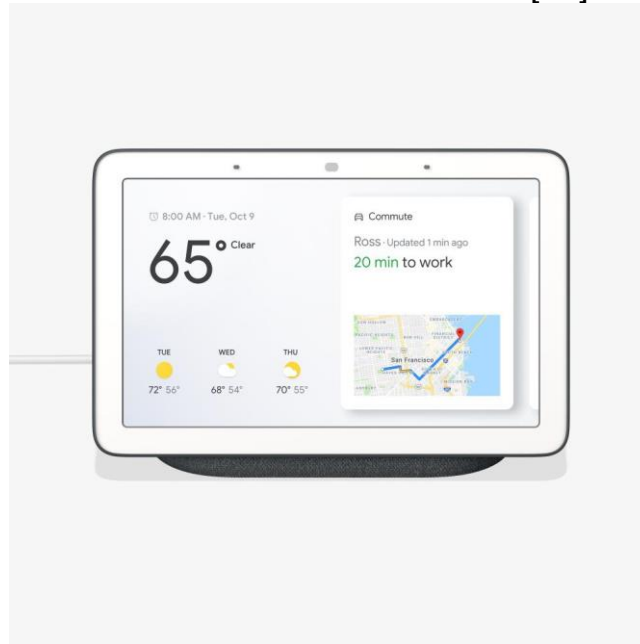
También hay paquetes de adaptación en cuanto al control del clima en el hogar, como Nest Thermostat E, un termostato inteligente que permite el control del clima, está diseñado con características de ahorro de energía que aprende a través de tu rutina [28].

Los periféricos como cámaras, clima, puertas eléctricas, bombillas, cortinas, todas las cosas del hogar conectadas al tendido eléctrico y con conectividad pueden ser controlados desde el cerebro del hogar inteligente, el “sistema central coordinador”.

Nest Hub Charcoal es un dispositivo tipo tablero como se muestra en la Ilustración 2.23, un asistente virtual capaz de facilitar el uso de las cosas del hogar con manos libres a través de la voz, puede controlar cámaras y



televisores compatibles a través de un solo tablero [29].



**Ilustración 2.23 Nest Hub**

La compañía Amazon tiene a Alexa la cual se observa en Ilustración 2.24, que es un asistente virtual que funge como sistema central coordinador de los dispositivos dentro del hogar, incluye la programación de rutinas y las sentencias son a través de la voz, por ejemplo, con solo decir “Alexa, prende la luz en la cocina”, “Alexa baja la intensidad de las luces en la recamara”, “Alexa, pon la sala en azul” [30].



**Ilustración 2.24 Alexa echo studio**

A su vez Apple y Google lanzaron un dispositivo bocina como se puede observar en la Ilustración 2.25, son utilizadas para controlar accesorios

inteligentes del hogar a través de la voz, que funcionan bajo el mismo paradigma de coordinación de los nodos del hogar. [31]



**Ilustración 2.25 Asistente virtual**

Como es observable existen algunas aplicaciones tecnológicas para la asistencia en el proceso de comunicación como las aplicaciones de IrisBound de la cual se hizo mención anteriormente, esta tecnología da soporte en el proceso de comunicación, adaptando el uso de la computadora para personas que no pueden hacer uso del mouse de una manera convencional.

Artículos recientes se han publicado en relación del hogar inteligente mediante el seguimiento de la mirada y la necesidad que existe en relación a la inclusión de personas con problemas de movilidad, sin embargo, aún no contemplan el proceso de comunicación ni la adaptación física a las personas con ELA [32].

# 3 Análisis y diseño del sistema

## 3.1 Introducción

En este capítulo se describe el análisis y diseño del sistema, que permitirá realizar las funcionalidades necesarias para alcanzar el objetivo de este sistema. Estas funcionalidades se llevarán a cabo usando la arquitectura del sistema que se describe en la sección 3.2, en la cual se explican los elementos de esta arquitectura. Posteriormente se describe el análisis y diseño del sistema bajo la metodología UML. En las secciones correspondientes se muestran los diversos diagramas del diseño UML. Finalmente, se presenta el diseño de la base de datos usado en el sistema.

## 3.2 Arquitectura del sistema

Básicamente el sistema usará una tableta para el control del hogar inteligente mediante un dispositivo de seguimiento de mirada.

El software desplegará al usuario un menú de navegación en la tableta, la cual a su vez proporcionará la posibilidad de coordinación a través de la red de wifi de los dispositivos dentro de la red. Así como la comunicación local mediante el uso de parlantes con palabras claves y el envío de mensajería instantánea mediante web API'S.

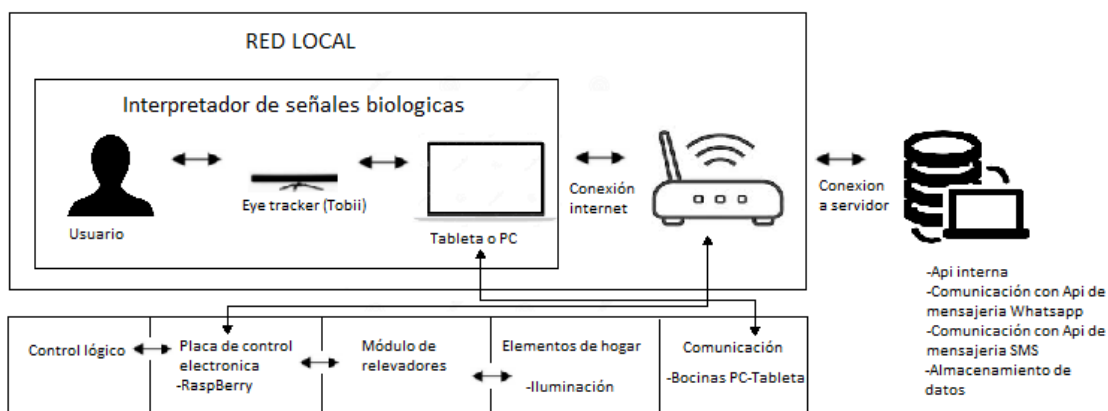


Ilustración 3.1 Arquitectura propuesta

En la Ilustración 3.1, se muestra la arquitectura propuesta del sistema, mediante el modelado de la integración de los diferentes módulos a construir. A continuación, se describirán los módulos generales a generar.

Primeramente, describiremos los componentes que se desarrollan dentro de la red local, los cuales son el usuario que utilizará el sistema, el eye tracker que controla las operaciones mediante la vista del usuario, la tableta o PC en donde está instalado el sistema que se encargará de la comunicación de los módulos y sus características de funcionamiento a través de conexión a internet o directamente con el hardware de la tableta o PC. Los componentes: usuario, eye tracker y tableta o PC son el interpretador de señales biológicas en conjunto.

**Placa de control electrónica:** Las señales que interpretará serán recibidas mediante la red de internet, las cuales serán enviadas por el sistema instalado en la tableta o PC, dentro de este módulo tenemos el **control lógico** de las operaciones tales como prender un foco, y el **módulo de relevadores** para el control físico del procedimiento.

**Elementos del hogar:** Se establece el control de los dispositivos como lo son los focos, bombillas, lámparas que tienen integrado un módulo IOT, el usuario fija su mirada en una opción y después de un cierto tiempo se despliega la acción como lo es prender y apagar, se manda la petición de la tableta al servidor interno que como coordinador manda la petición al dispositivo, que a su vez encenderá el dispositivo.

**Comunicación:** En este módulo se le desplegará al usuario la opción de poder enviar mensajes SMS así como Whatsapp, a un listado de contactos, Esto con ayuda del teclado habilitado de control ocular de w10 y con la ayuda de consumo de api's de terceros. Además, está el uso y función de las bocinas del dispositivo donde está instalado el sistema.

## 3.3 Diseño del Sistema

4+1 es un modelo diseñado por Philippe Kruchten para "describir la arquitectura de sistemas de software, basados en el uso de múltiples vistas concurrentes". Las vistas suelen describir el sistema desde el punto de vista de diferentes interesados, tales como usuarios finales, desarrolladores o directores de proyecto.

Las cuatro vistas del modelo son: vista lógica, vista de implementación, vista de proceso y vista física. Además, una selección de casos de uso o escenarios suele utilizarse para ilustrar la arquitectura sirviendo como una vista más. Por ello el modelo contiene 4+1 vistas:

- Vista lógica: La **vista lógica** está enfocada en describir la estructura y funcionalidad del sistema. Los diagramas UML que se utilizan para representar esta vista son los Diagrama de Clase
- Vista de desarrollo: La **vista de desarrollo** ilustra el sistema de la perspectiva del programador y está enfocada en la administración de los artefactos de software. Esta vista también se conoce como vista de implementación. Utiliza el Diagrama de Componentes UML para describir los componentes de sistema.
- Vista de proceso: La **vista de proceso** trata los aspectos dinámicos del sistema, explica los procesos de sistema y cómo se comunican. Se enfoca en el comportamiento del sistema en tiempo de ejecución. La vista considera aspectos de concurrencia, distribución, rendimiento, escalabilidad, etc.
- Vista física: La **vista física** describe el sistema desde el punto de vista de un ingeniero de sistemas. Está relacionada con la topología de componentes de software en la capa física, así como las conexiones físicas entre estos componentes. Esta vista también se conoce como vista de despliegue. En UML se utiliza el Diagrama de Despliegue para representar esta vista.
- Escenarios: La descripción de la arquitectura se ilustra utilizando un conjunto de casos de uso, o **escenarios** lo que genera una quinta vista. Los escenarios describen secuencias de interacciones entre objetos, y entre procesos. Se utilizan para identificar y validar el diseño de arquitectura. Esta vista es también conocida como vista de casos de uso.

### 3.3.1 Diagrama de vista de procesos

Basándose en la definición de un diagrama de secuencia, la cual nos dice que este tipo de diagrama nos muestra la interacción de objetos en una aplicación a través del tiempo, se hizo un análisis general de cómo representar dicho diagrama y el resultado fue el diagrama de la Ilustración 3.2, el cual muestra la interacción de un usuario final con los casos de uso del sistema

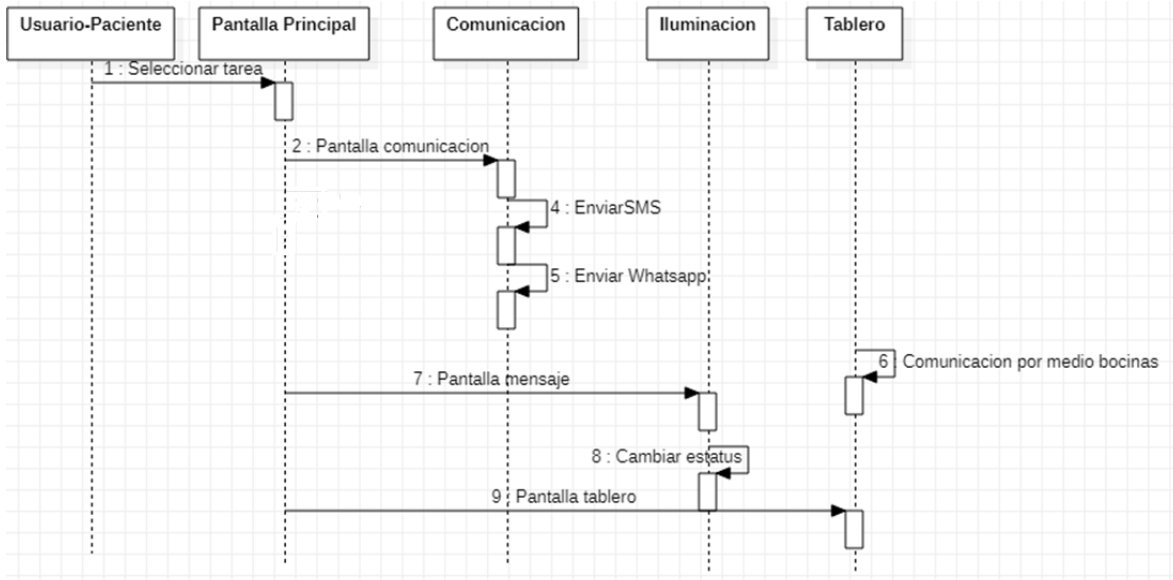


Ilustración 3.2 Diagrama de secuencia

### 3.3.2 Diagrama de vista Lógica

Teniendo el conocimiento de la definición de un diagrama de clases el cual nos dice que este tipo de diagrama describe la estructura de un sistema mostrando las clases, atributos, métodos y las relaciones entre los objetos, se realizó una estructura general para guiarse durante el desarrollo como se muestra en la Ilustración 3.3.

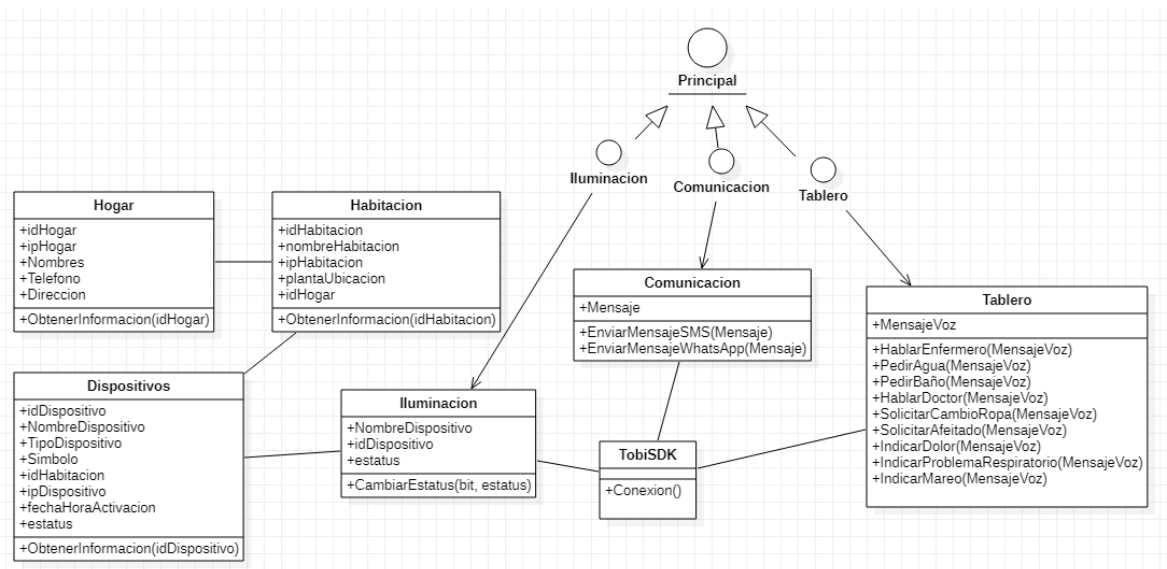


Ilustración 3.3 Diagrama de vista lógica

### 3.3.3 Diagrama de implementación

En esta sección se añadió el diagrama de componentes necesario, para visualizar la interacción en un aspecto general del sistema, los componentes principales como su Interfaz y clases, el modelo de base de datos y sus archivos correspondientes. El resultado que se obtuvo en el desarrollo de dicho diagrama fue el desplegado en la Ilustración 3.4.

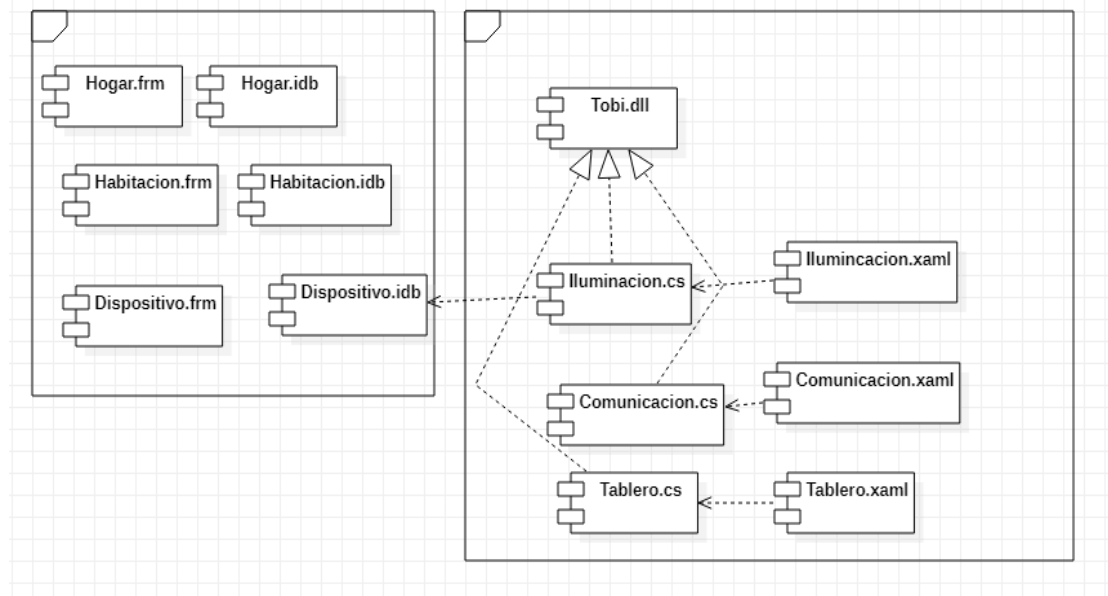
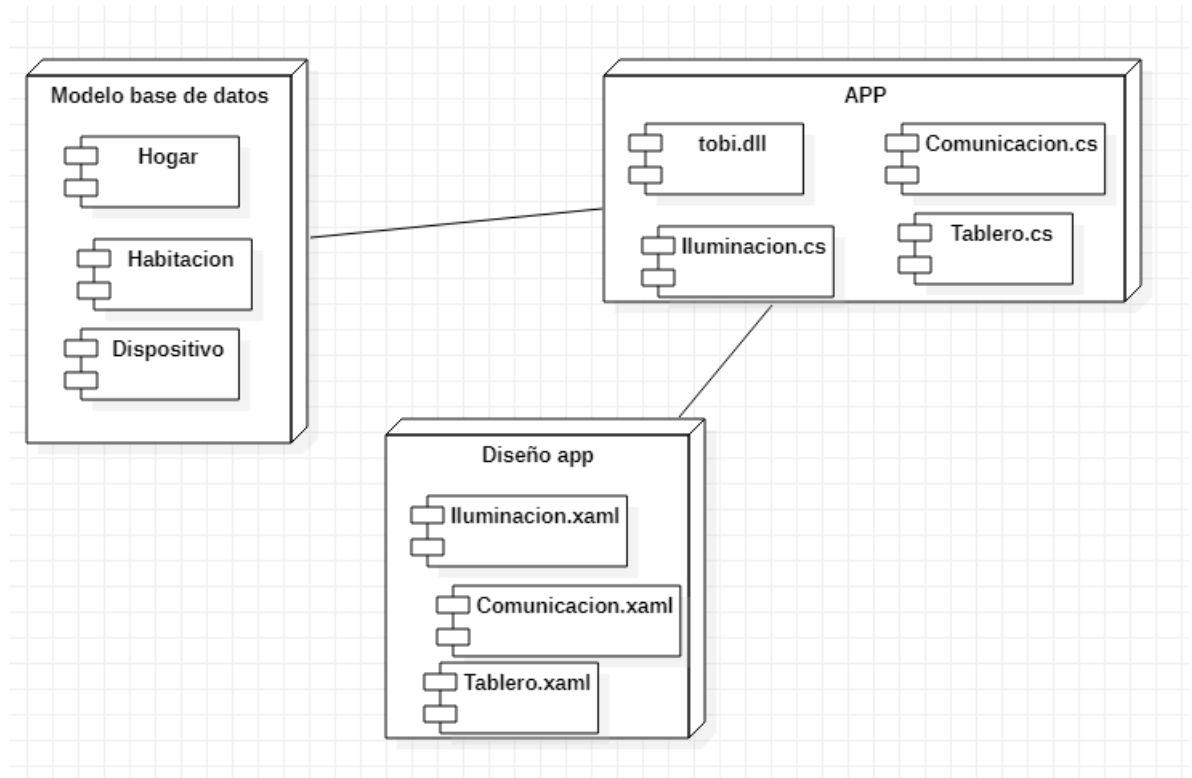


Ilustración 3.4 Diagrama de implementación

### 3.3.4 Diagrama de vista de despliegue

En esta vista se representó mediante nodos los diferentes puntos que interactúan para el funcionamiento del sistema, que a grandes rasgos es la definición de un diagrama de despliegue cuyo resultado fue el que se muestra la Ilustración 3.5.



**Ilustración 3.5 Diagrama de vista de despliegue**

### 3.3.5 Diagrama de vista de escenarios

En esta vista se realizó los casos de uso del sistema para el cliente/usuario final, es un diagrama general que incluye todos los procesos a realizar dentro del sistema como se muestra en la Ilustración 3.6.



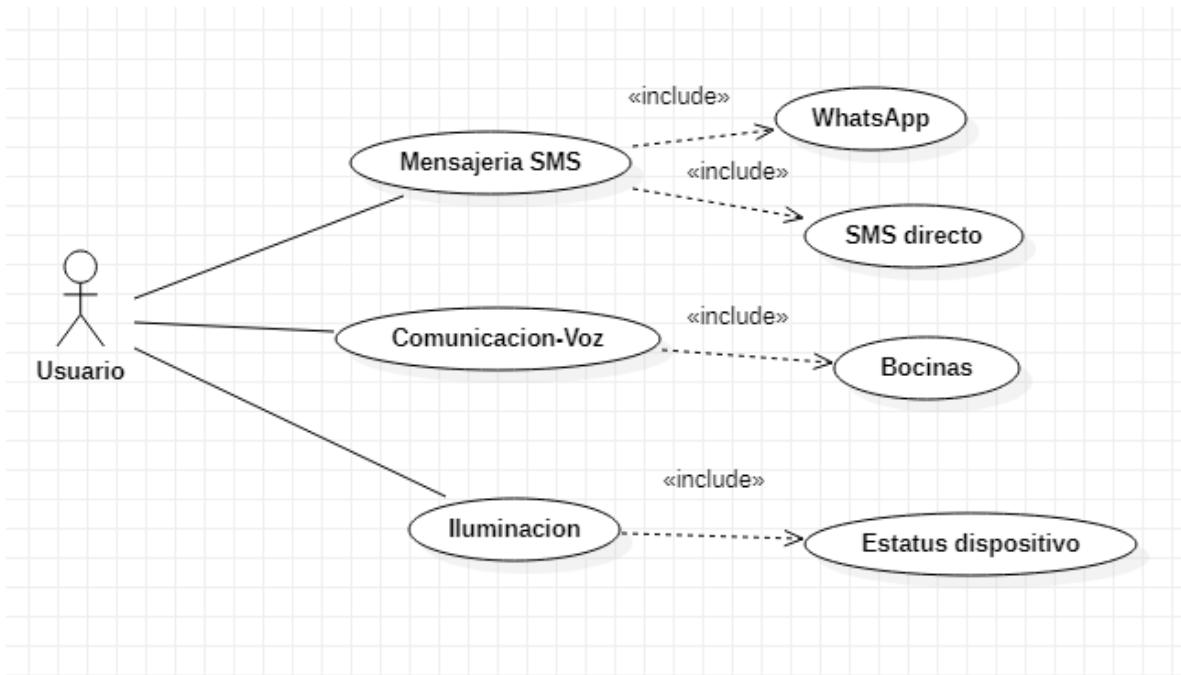


Ilustración 3.6 Diagrama de vista de escenarios

### 3.3.6 Especificaciones de casos de uso

En esta sección se presentan los diferentes casos de uso que se usan para el diseño del sistema.

Nombre	Iluminación IOT
<b>Descripción:</b> Permite controlar la iluminación en las habitaciones del hogar mediante el internet de las cosas.	
<b>Condición:</b> Instalación de la arquitectura necesaria en el hogar para la conexión con el sistema.	
<b>Actores:</b> Usuario	
<b>Flujo normal:</b>	
<ol style="list-style-type: none"> <li>1. El usuario fija mirada en pestaña de iluminación.</li> <li>2. El usuario selecciona la habitación a la cual se manipulará la iluminación.</li> <li>3. El usuario envía cambios de iluminación.</li> <li>4. El sistema envía petición a placa electrónica de control para el control del módulo de relevadores y a su vez realizar la tarea del control de iluminación.</li> </ol>	

5. El sistema envía mensaje exitoso al completar operación.
<b>Flujo Alternativo:</b>
1. El usuario cambia de pestaña y no realiza control de iluminación.
2. El sistema manda mensaje de error al no completar la tarea.

<b>Nombre</b>	Mensajería SMS
<b>Descripción:</b> Permite enviar mensajes a contactos especificados en el sistema, como enfermeros, doctores, etc....	
<b>Condición:</b> Acceso a internet.	
<b>Precondición:</b> Calibración de sensor ocular.	
<b>Actores:</b> Usuario	
<b>Flujo normal:</b>	
1. El usuario fija mirada en pestaña de comunicación.	
2. El usuario selecciona contacto.	
3. El usuario tiene opción de escribir mensaje en un formulario mediante el teclado virtual de la tableta o PC o seleccionar un mensaje.	
4. El usuario envía mensaje seleccionando que tipo, WhatsApp o SMS directo.	
5. El sistema mediante API'S realiza la operación de envío de mensajes.	
6. El sistema retorna mensaje de envió exitoso.	
<b>Flujo Alternativo:</b>	
1. El usuario cambia de pestaña y no envía mensaje.	
2. El sistema manda mensaje de error al no enviar el mensaje.	

<b>Nombre</b>	Comunicación-Voz
<b>Descripción:</b> Permite comunicación de voz mediante las bocinas de la tableta o PC, en los cuales la aplicación está instalada.	
<b>Condición:</b> Bocinas de tableta o PC existente.	
<b>Precondición:</b> Configurar volumen de bocinas.	
<b>Actores:</b> Usuario	
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. El usuario fija mirada en pestaña de tablero.</li> <li>2. El usuario selecciona por medio de los diseños en tablero el tipo de mensaje de voz.</li> <li>3. El usuario tiene opción de escribir mensaje en un formulario mediante el teclado virtual de la tableta o PC para que sea interpretado por el sistema y enviado a las bocinas o seleccionar un mensaje de voz del tablero.</li> <li>4. El usuario envía mensaje y se reproduce en bocinas como audio.</li> <li>5. El sistema reproduce mensaje de audio en bocinas.</li> </ol>	
<b>Flujo Alternativo:</b> <ol style="list-style-type: none"> <li>1. El usuario cambia de pestaña y no envía mensaje de voz.</li> <li>2. El sistema manda mensaje de error al no enviar el mensaje de voz.</li> </ol>	

## 3.4 Diseño de la base de datos

### 3.4.1 Diseño conceptual (Modelo ER)

En la Ilustración 3.7, se muestra el diagrama ER del diseño conceptual de la base de datos. Este modelo Entidad – Vinculo (ER) se basa en una percepción de un mundo real que consiste en un conjunto de objetos básicos llamados entidades, y de interrelaciones entre estos objetos. Está pensado como una notación orientada al diseño del esquema conceptual. Permite la descripción del esquema conceptual, sin preocuparse por problemas de diseño físico o eficiencia. En una etapa posterior, el diagrama ER es llevado al modelo relacional.

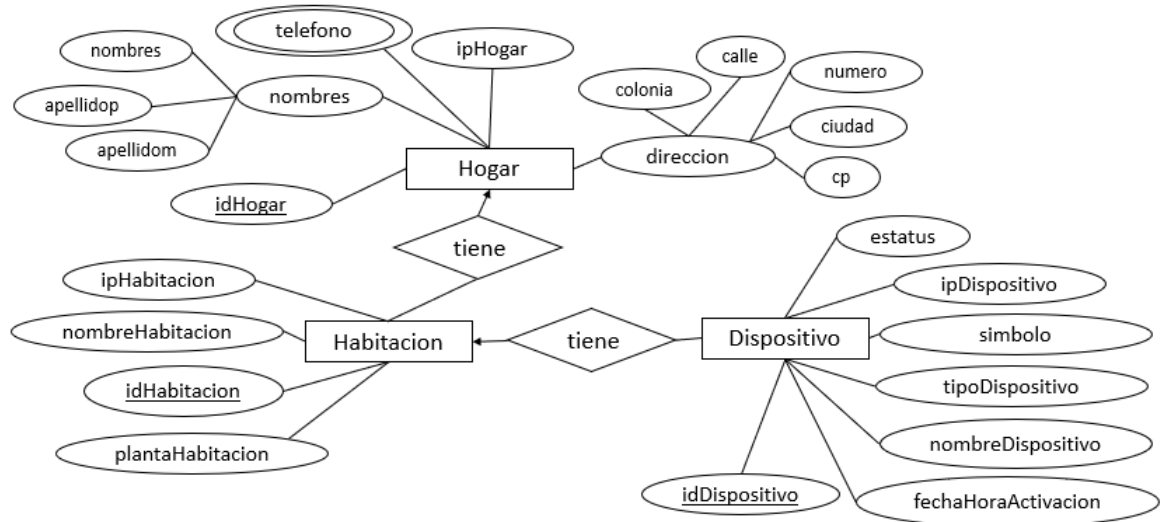
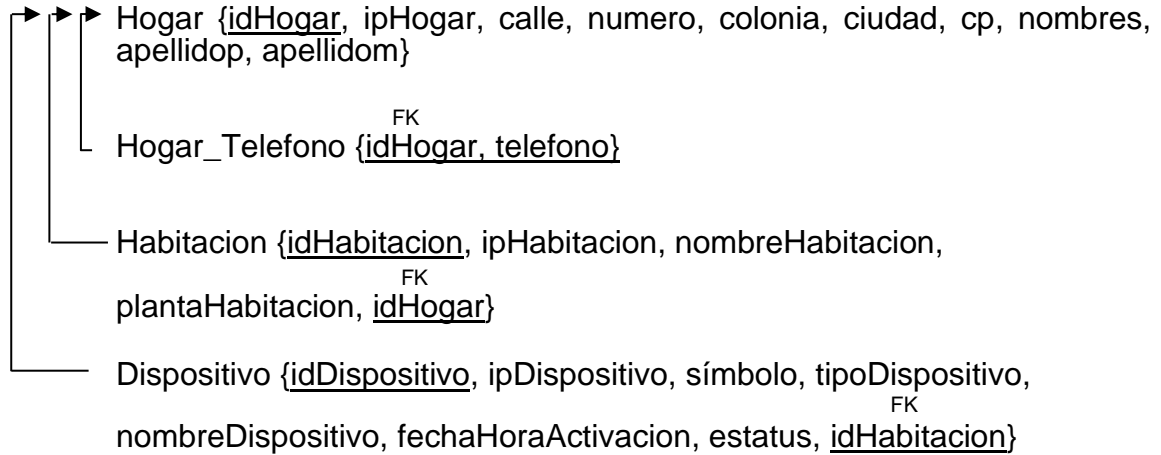


Ilustración 3.7 Diagrama entidad relación

### 3.4.2 Transformación al modelo Relacional

En el modelo relacional los datos y las interrelaciones entre los datos se presentan por medio de un conjunto de tablas, cada una de las cuales tiene varias columnas con nombres únicos y una columna de una tabla representa una relación entre un conjunto de valores. Se puede trasladar un esquema ER a un esquema relacional siguiendo los pasos del algoritmo descrito en este texto<sup>1</sup>.

El resultado de la transformación es presentado a continuación:



### 3.4.3 Normalización

La primera forma normal (1FN), para que se cumpla tiene como característica las siguientes reglas:

- Eliminar grupos repetidos en tablas individuales.
- Crear una tabla independiente para cada conjunto de datos relacionados.
- Identificar cada conjunto de relacionados con la clave principal.

No se utilizaron varios campos en una sola tabla para almacenar datos similares.

Tabla Habitaciones no normalizada:

idHogar	ipHogar	nombreHabitacion1	nombreHabitacion2
777	100.100.100.100	Sala	Comedor

Normalizada primera forma:

idHogar	idHabitacion	nombreHabitacion
777	888	Sala
777	999	Comedor

La regla para la Segunda Forma Normal (2FN) asegura que cada tributo describe la entidad, es decir crear tablas separadas y relacionarlas con una clave externa (Foreign Key).

Tabla hogar:

<b>idHogar</b>	<b>ipHogar</b>
777	100.100.100.100

Tabla Habitaciones:

<b>idHabitacion</b>	<b>nombreHabitacion</b>	<b>idHogar(Llave Ext.)</b>
888	Sala	777
999	Comedor	777

### 3.4.4 Implementación de la base de datos

Descripción del esquema de la base de datos.

#### **Hogar**

idHogar → Int primary key

IpHogar → varchar (15)

Nombres: → varchar (30)

Apellidoop → varchar (30)

Apellidoom → varchar (30)

Calle → varchar (30)

Numero → int

Colonia → varchar (30)

Ciudad → varchar (30)

Cp → int

#### **Habitación**

idHabitacion → int primary key

IpHabitación → varchar (15)

nombreHabitacion → varchar (30)

plantaHabitacion → varchar (20)

idHogar → int foreign key

#### **Dispositivo**

idDispositivo → Int primary key

ipDispositivo → varchar (15)

nombreDispositivo → varchar (30)

fechaHoraActivacion → datetime

Estatus → bit

idHabitacion → int foreign key

## HogarTelefono

idHogar → int foreign key

Telefono → char(10) primary key

En el diagrama anterior se muestra el esquema de las tablas de la base de datos, el cual será utilizado para el renderizado de las habitaciones y dispositivos en la aplicación tableta.

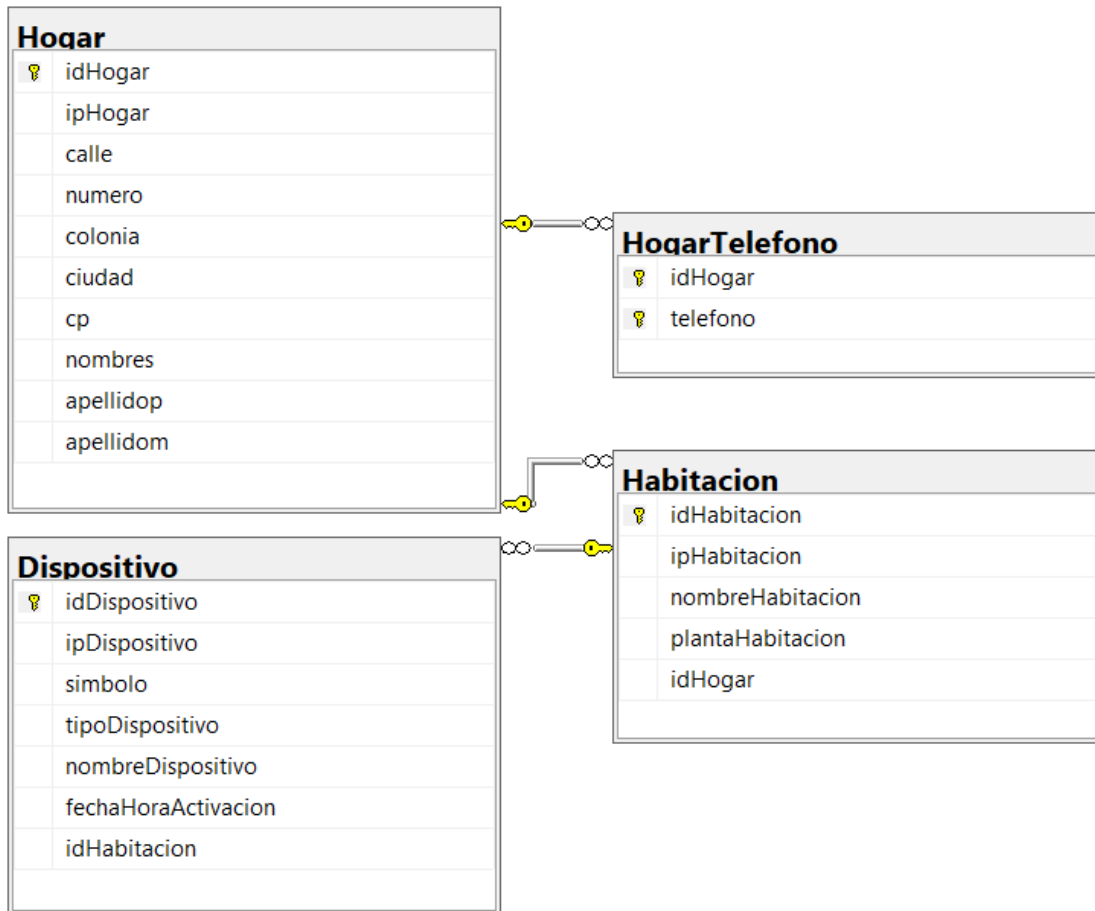


Ilustración 3.8 Diagrama de base de datos

En la Ilustración 3.8 se puede observar la distribución de la base de datos, la tabla nombrada habitación tiene información del nombre, planta, hogar al que pertenece (idHogar), entre otros. Está relacionada con la tabla dispositivos y la de hogar, en la tabla de dispositivos está contenida como llave foránea y además tiene llave foránea de la tabla hogar. En la tabla hogar podemos observar los campos relacionados a la información del usuario en sí, tales como su teléfono y dirección; a su vez, la tabla dispositivos almacena un campo estatus del dispositivo, para informar en caso de ser necesario el último estado registrado, ya sea apagado o prendido.

## 4 Implementación del sistema

Mediante el análisis previamente realizado, se identificaron los requisitos y necesidades del sistema propuesto y se determinó e implementó la mejor solución, a la medida de las necesidades del proyecto.

El sistema para tableta de automatización del hogar se desarrolló en el framework de .NET, utilizando el lenguaje de programación de C# en el Back-End, para el diseño Front-End el lenguaje de XAML, en el entorno de desarrollo de Visual Studio y para el seguimiento de mirada se empleó el dispositivo de tobii Eye Tracking y su software principal Tobii Eye Tracking v2.16.5.

Como coordinador en el hogar para el control de las cosas, se utilizó una computadora de bajo costo Raspberry con el sistema operativo de Raspbian, el cual tiene embebido un webservice del microframework flask de Python.

Las cosas del hogar como las luces fueron prototipadas en Arduino, la función de comunicación mediante el envío de mensajería se hizo a través de API'S de terceros y se utilizó la librería de System.Speech.Synthesis para reproducir mediante el micrófono las palabras de acceso rápido.

A continuación, se presentará la interfaz tableta utilizada para la interacción con las cosas del hogar mediante el seguimiento de mirada, la configuración del coordinador donde se hospeda el servicio web y la comunicación con las cosas del hogar y las API'S de terceros.



# 4.1 Interfaz del sistema

Para poder hacer uso de la aplicación y obtener una mejor experiencia de usabilidad es necesario calibrar el sensor a la medida del usuario que estará utilizando el sistema a través del instalador de Tobii. En la Ilustración 4.1 Gestor de perfil de usuarios se puede observar el panel para crear un perfil de usuario.

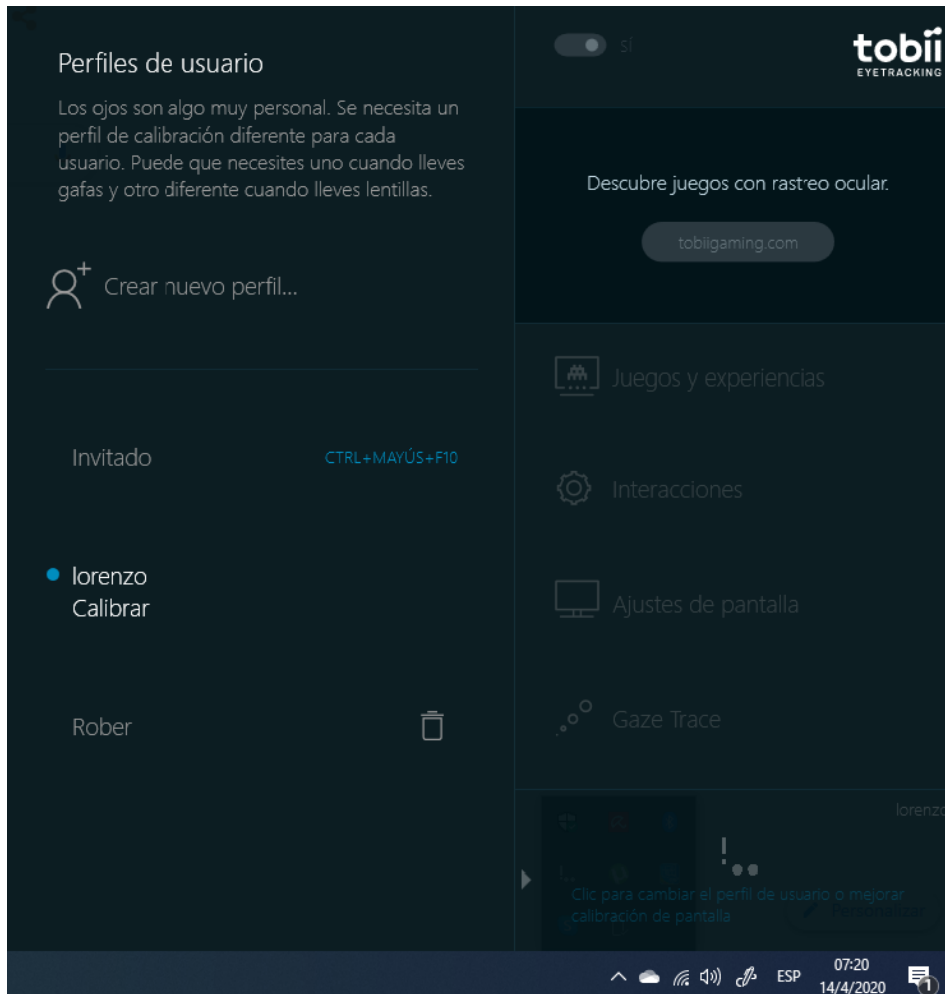
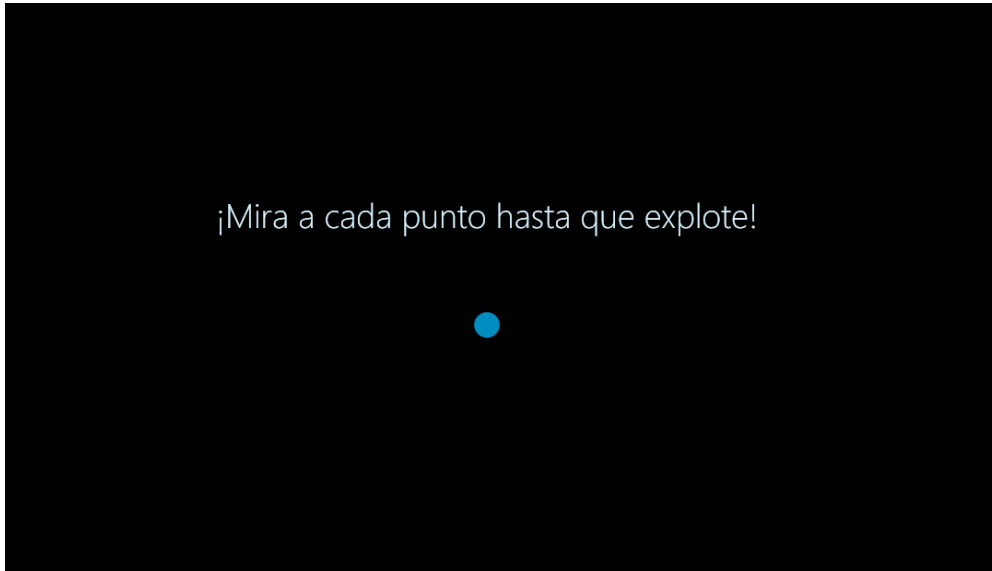


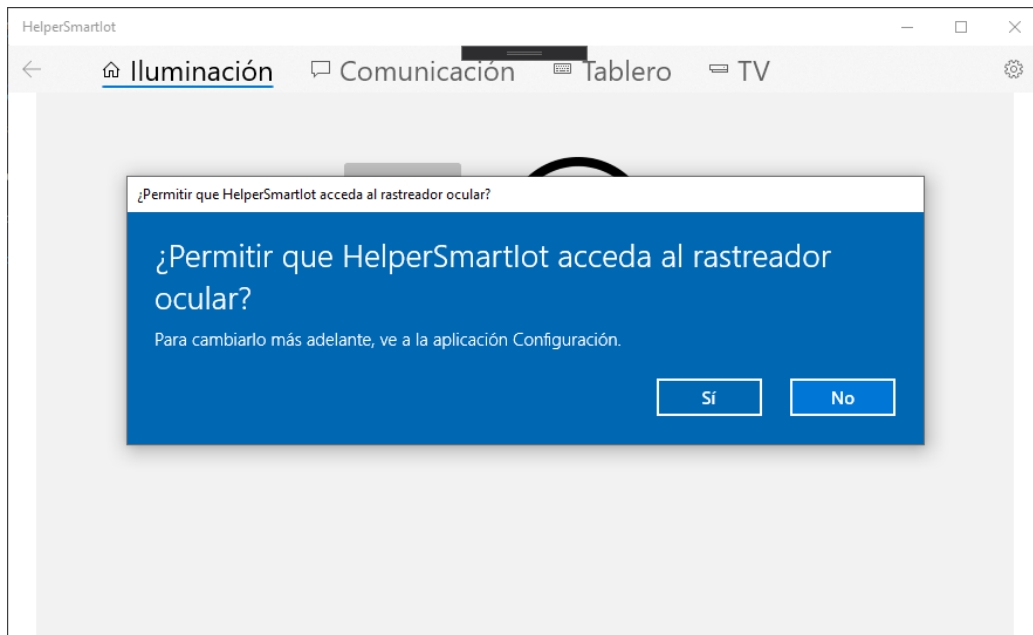
Ilustración 4.1 Gestor de perfil de usuarios

Al crear el perfil de usuario se establece una calibración a la medida por usuario, a través de proyecciones de puntos que vinculan la interfaz con el sensor, como se puede observar en la Ilustración 4.2 Calibración de dispositivo.



**Ilustración 4.2 Calibración de dispositivo**

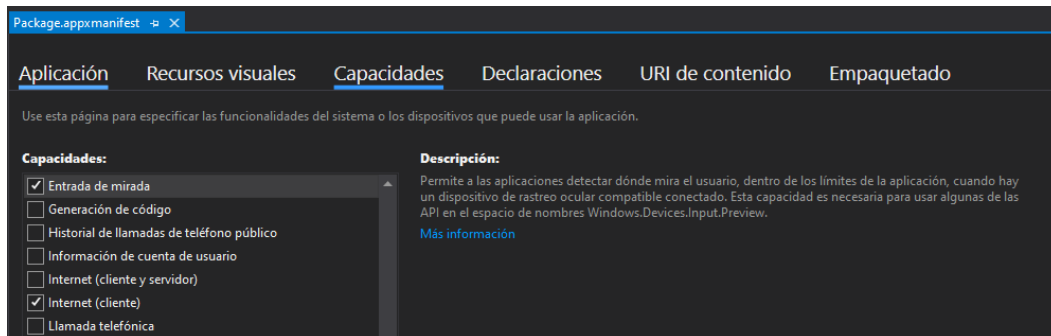
Una vez calibrado el sensor, la primera vez que se ejecuta la aplicación en la tableta te pedirá permiso para acceder al uso del dispositivo de seguimiento de mirada como se muestra en la Ilustración 4.3.



**Ilustración 4.3 Permisos seguimiento ocular**

Para poder hacer uso del sensor se tiene que indicar en el manifiesto del proyecto como se muestra en la Ilustración 4.4, al igual que cualquier otro

sensor que necesitara el sistema, como requisito funcional se debe indicar en esta parte como uso de la cámara, micrófono, etc.



**Ilustración 4.4 Manifiesto**

Se diseñó una interfaz para permitir al usuario con ELA la interacción con los dispositivos del hogar mediante una interfaz amigable a través del SO como se muestra en la Ilustración 4.5 Pantalla principal donde se puede observar la pantalla de inicio de la aplicación.



**Ilustración 4.5 Pantalla principal**

Para la funcionalidad de desplazamiento se utiliza una barra de progreso circular como se muestra en Ilustración 4.6 Radial progress bar, al momento de fijar la mirada en un punto determinado de la interfaz donde se encuentra algún control, como un botón, por un tiempo determinado se inicializa un contador para desencadenar el evento deseado.

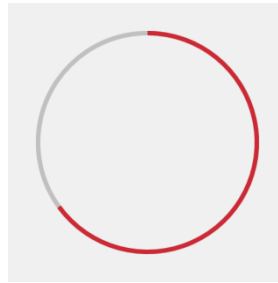


Ilustración 4.6 Radial progress bar

La integración de la barra de progreso se realiza en el XAML de la vista donde opera esta funcionalidad, se hizo uso de los componentes de la utilería de UWP community Toolkit. En la Ilustración 4.7 estructura xaml vista principal, se integra un radial progressbar y una elipse con sus respectivos atributos.

```
<Canvas x:Name="gazePositionCanvas" Grid.Row="1" >
  <my:RadialProgressBar
x:FieldModifier="public"
x:Name="GazeRadialProgressBar"
  Value="0"
  RequestedTheme="Light"
  Foreground="Black"
  Thickness="4"
  Minimum="0"
  Maximum="100"
  Width="20"
  Height="20"
  Outline="White"
  Visibility="Collapsed"/>
  <Ellipse
    x:Name="eyeGazePositionEllipse"
    Width="20" Height="20"
    Fill="Blue"
    Opacity="0.5"
    Visibility="Collapsed"/>
</Canvas>
```

Ilustración 4.7 estructura xaml vista principal

A continuación, como se muestra en la Ilustración 4.8, se declaran las variables necesarias para dar funcionalidad al seguimiento de mirada en la interfaz gráfica.

```

/// <summary>
/// Referencia a los ojos y la cabeza del usuario según lo detectado
/// por el dispositivo de seguimiento ocular.
/// </summary>
private GazeInputSourcePreview fuenteMirada;
/// <summary>
/// Recibe notificaciones de eventos cuando se agrega un dispositivo.
/// </remarks>
private GazeDeviceWatcherPreview dispositivoMiradaObs;
/// <summary>
/// Temporizador para enfocar la mirada en RadialProgressBar.
/// </summary>
public static DispatcherTimer timerGaze = new DispatcherTimer();
/// <summary>
/// El rastreador se usa para evitar que el temporizador de mirada se reinicie.
/// </summary>
public static bool timerStarted = false;

```

Ilustración 4.8 Variables globales

Nuestro panel de menú consta por varias funcionalidades en sus contenedores, para la navegación entre el menú mediante el seguimiento de mirada se inicializa la funcionalidad con el método `IniciarObservadorMirada` en el constructor del menú principal.

```

1 referencia
private void Initialize()
{
    NavigationService.Frame = shellFrame;
    NavigationService.NavigationFailed += Frame_NavigationFailed;
    NavigationService.Navigated += Frame_Navigated;
    navigationView.BackRequested += OnBackRequested;
    IniciarObservadorMirada();//activar seguimiento
}

```

Ilustración 4.9 Inicialización seguimiento de mirada

Mediante el método `IniciarObservadorMirada` se crea un objeto para supervisar el estado del dispositivo de SO vinculado a la vista actual y se declaran los eventos asociados al objeto, tal y como se muestra en la Ilustración 4.10.

```

1 referencia
private void IniciarObservadorMirada()
{
    if (dispositivoMiradaObs == null)
    {
        dispositivoMiradaObs = GazeInputSourcePreview.CreateWatcher();
        dispositivoMiradaObs.Added += this.DispositivoAgregado;
        dispositivoMiradaObs.Start();
    }
}

```

Ilustración 4.10 Observador mirada

Se le agrega al objeto creado una configuración inicial, como el intervalo de tiempo del temporizador de mirada una vez detectado el dispositivo, se inicializa enlazando el seguimiento a la aplicación activa, como se observa en la Ilustración 4.11.

```

private void ConfiguracionSeguimiento(GazeDevicePreview gazeDevice)
{
    if (DispositivoDisponible(gazeDevice))
    {
        timerGaze.Interval = new TimeSpan(0, 0, 0, 0, 0); //timer set
        timerGaze.Tick += TimerGaze_Tick;
        fuenteMirada = GazeInputSourcePreview.GetForCurrentView();
        fuenteMirada.GazeMoved += MiradaMov;
    }
}

```

Ilustración 4.11 configuración Seguimiento

Antes de establecer estos valores, se debe verificar si el dispositivo se encuentra disponible, mediante las instrucciones de la Ilustración 4.12 se puede detectar si los ojos del usuario se encuentran detectados y se puede realizar el seguimiento de los mismos.

```

private bool DispositivoDisponible(GazeDevicePreview gazeDevice)
{
    return (gazeDevice.CanTrackEyes &&
        gazeDevice.ConfigurationState ==
        GazeDeviceConfigurationStatePreview.Ready);
}

```

Ilustración 4.12 Estatus Dispositivo

El control de progreso se superpone y restablece cada vez que se fija la mirada en las coordenadas del control que se busca desencadenar. La elipse de la barra de progreso se define una vez que se vincula y detecta que la vista está fija en el control, estos parámetros se definen como se muestra en la Ilustración 4.13.

```

TranslateTransform translateTarget = new TranslateTransform();
1 referencia
private void SetGazeTargetLocationTablero(UIElement element)
{
    var ttv = element.TransformToVisual(Window.Current.Content);
    Point screenCoords = ttv.TransformPoint(new Point(0, 0));
    translateTarget.X = (int)screenCoords.X + 50;
    translateTarget.Y = (int)screenCoords.Y;
    GazeRadialProgressBar.RenderTransform = translateTarget;
}

```

Ilustración 4.13 Definición Elipse

En el evento según la Ilustración 4.14, se valida que se encuentre el punto de mirada detectado, una vez obtenidas las coordenadas, se realiza un match con el control que se encuentran fijos los ojos, si el usuario se encuentra mirando para inicializar posteriormente el llenado de la misma a través del contador de progreso.

```

bool MenuTablero;
1 referencia
private void MiradaMov(GazeInputSourcePreview sender, GazeMovedPreviewEventArgs args)
{
    if (args.CurrentPoint.EyeGazePosition != null)
    {
        double gazePointX = args.CurrentPoint.EyeGazePosition.Value.X;
        double gazePointY = args.CurrentPoint.EyeGazePosition.Value.Y;
        Point gazePoint = new Point(gazePointX, gazePointY);
        bool hitRadialProgressBar = true;
        MenuTablero = DoesElementContainPoint(
            gazePoint,
            GazeRadialProgressBar.Name,
            TableroMenu);
        GazeRadialProgressBar.Visibility = Visibility.Visible;
        if (MenuTablero)
        {
            SetGazeTargetLocationTablero(TableroMenu);
        }
        if (hitRadialProgressBar)
        {
            GazeRadialProgressBar.Thickness = 10;
        }
        else
        {
            GazeRadialProgressBar.Thickness = 4;
        }
        args.Handled = true;
    }
}

```

Ilustración 4.14 Conteo Progreso

Una vez ubicado el elemento en el control se inicializa el contador como se observa en el método de la Ilustración 4.15, para detectar el tiempo que permanece fija la mirada y desencadenar el evento deseado, el temporizador se inicializa cuando la mirada se encuentra fija en el control dentro del contorno de la barra de progreso.

```
3 referencias
private bool DoesElementContainPoint(Point gazePoint, string elementName, UIElement uiElement)
{
    IEnumerable<UIElement> elementStack =
        VisualTreeHelper.FindElementsInHostCoordinates(gazePoint, uiElement, true);
    foreach (UIElement item in elementStack)
    {
        if (item is FrameworkElement feItem)
        {
            if (!timerStarted)
            {
                timerGaze.Start();
                timerStarted = true;
            }
            return true;
        }
    }
    return false;
}
```

Ilustración 4.15 Temporizador de mirada

Posteriormente, una vez que la barra de progreso está completada se realiza el cambio de menú como se muestra en la Ilustración 4.16

```
1 referencia
private void TimerGaze_Tick(object sender, object e)
{
    GazeRadialProgressBar.Value += 1;
    if (GazeRadialProgressBar.Value == 100)
    {
        if (MenuLuces)
        {
            CambiarMenu("Iluminación");
        }
        else if (MenuComunicacion)
        {
            CambiarMenu("Comunicacion");
        }
        else if (MenuTablero)
        {
            CambiarMenu("Tablero");
        }
    }
}
```

Ilustración 4.16 Despliegue de Menú



## 4.2 Triangulación de comunicación

En el sistema se realiza una petición haciendo uso de la librería System.Net.Http desde la host tableta hacia el Raspberry, donde se encuentra alojada el web API con el microframework de Python flask, la siguiente Ilustración 4.17 muestra la configuración.

```
#region Encender Luz
1 referencia
private async Task encenderLuzHabitacionAsync()
{
    using (var client = new HttpClient())
    {
        var response = await client.GetAsync("192.168.1.91:8081/encenderLuz");
        var statusCode = response.StatusCode;
        if ((int)statusCode == 200 && response.IsSuccessStatusCode) {...}
    }
}
#endregion
```

Ilustración 4.17 Petición al Coordinador

Se debe tener en cuenta la configuración inicial del app.run de la web API para configurar la dirección estática que tendrá dentro de la red, tal y como se muestra en la Ilustración 4.18.

```
if __name__ == '__main__':
    HOST = environ.get('SERVER_HOST', 'localhost')
    try:
        PORT = int(environ.get('SERVER_PORT', '5555'))
    except ValueError:
        PORT = 5555
    #configuracion de direccion estatica en web API
    app.run('192.168.1.91', 8081)
```

Ilustración 4.18 Ip estática Web Api

Posteriormente el Raspberry le indica al componente “foco”, la acción a realizar, en este caso el encendido de la luz como se puede observar en la Ilustración 4.19, quien accionará el relevador según la lógica cargada en el sistema embebido,

```
@app.route('/encenderLuz', methods=['GET', 'POST'])
def encenderLuz():
    request_json = request.get_json("http://192.168.1.108/ledcontrol?ledstate=on");
    return json.dumps({'success':True}), 200, {'ContentType':'application/json'}
```

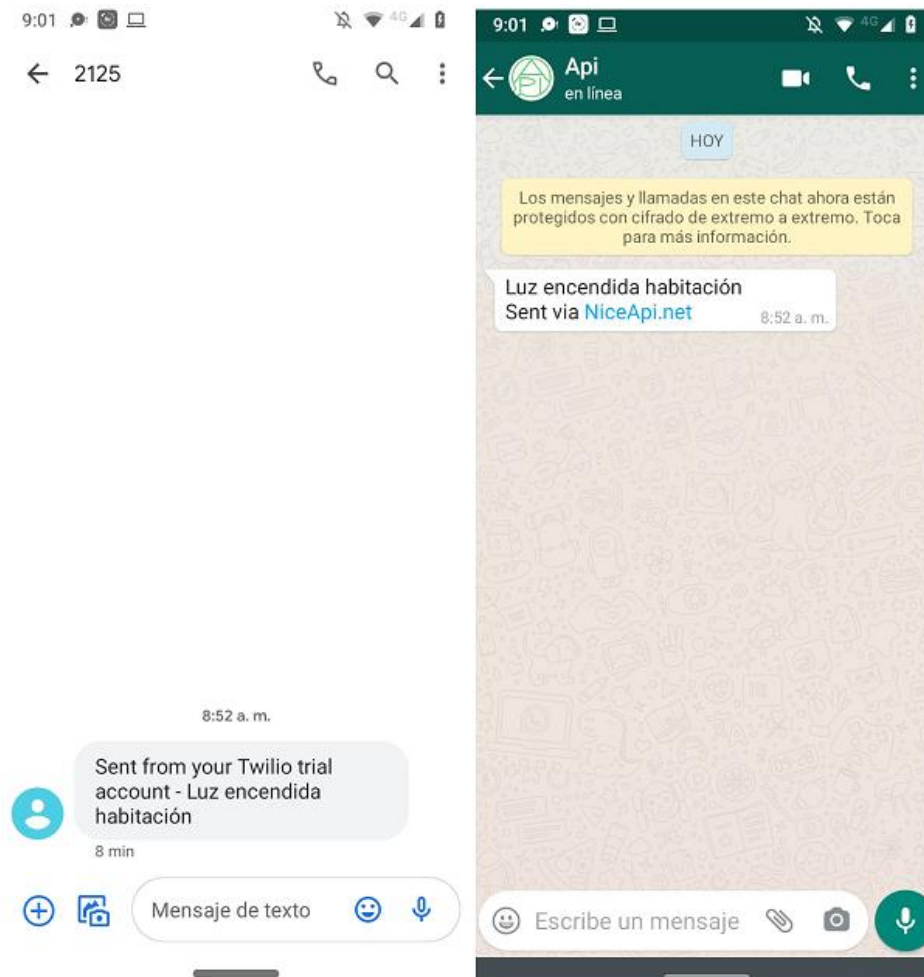
**Ilustración 4.19 Web API Controller**

Una vez encendido el foco, la web API responde a la tableta con un estatus, en caso de que la comunicación haya sido satisfactoria, se envía un mensaje personalizado a través de SMS y WhatsApp, a los números configurados previamente y se cambia el icono del foco a encendido como se muestra en la Ilustración 4.20.



**Ilustración 4.20 Icono Encendido**

A continuación, se muestra en la Ilustración 4.21 cómo se realiza el uso de web API'S de terceros como Twillo y NiceApi, para informar que ha sido encendida la luz de la habitación.



**Ilustración 4.21 API'S Mensajería de terceros**

El envío de mensajería se realiza desde la tableta, mediante el uso de las credenciales otorgadas por el proveedor del servicio, tal y como se muestra en la Ilustración 4.22.

```
string yourId = "your_id";
string yourMobile = "your_phone";
string yourMessage = "Luz encendida habitación";
const string accountSid = "your_accountSid";
const string authToken = "your_auth_token";
TwilioClient.Init(accountSid, authToken);
ServicePointManager.SecurityProtocol = (SecurityProtocolType)3072; //TLS 1.2

try
{
    var message = MessageResource.Create(
        body: yourMessage,
        from: new Twilio.Types.PhoneNumber("twillo_phone"),
        statusCallback: new Uri("http://postb.in/1234abcd"),
        to: new Twilio.Types.PhoneNumber("your_phone")
    );
    Console.WriteLine(message.Sid);
    string url = "https://NiceApi.net/API";
    HttpRequest request = (HttpRequest)WebRequest.Create(url);
    request.Method = "POST";
    request.ContentType = "application/x-www-form-urlencoded";
    request.Headers.Add("X-APIId", yourId);
    request.Headers.Add("X-APIMobile", yourMobile);
}
```

Ilustración 4.22 Configuración C# Mensajería

## 4.3 Control Iluminación

El control de iluminación se hace a través de un módulo de comunicación esp8266 y un relevador que tiene la tarea de permitir o denegar el paso de corriente hacia la bombilla, la programación fue subida al módulo a través del IDE Arduino.

En la Ilustración 4.23 se puede observar la asociación con la red donde estará localizada la bombilla, se realiza la conexión dentro de la misma y se definen las variables globales como el relevador y otras dependencias como las librerías de las que hará uso el módulo de comunicación para llevar a cabo su funcionalidad.

```

#include "ESP8266WiFi.h"
#include "ESP8266WebServer.h"

#define ssid "Linksys04364"
#define password "qgfhyxkmp"
#define relay 0
ESP8266WebServer server(80);
void setup()
{
  Serial.begin(115200);
  pinMode(relay, OUTPUT);
  digitalWrite(relay, LOW); // Initialize the LED_BUILTIN pin as an output

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print("*");
  }

  Serial.println("");
  Serial.println("WiFi connection Successful");
  Serial.print("The IP Address of ESP8266 Module is: ");
  Serial.print(WiFi.localIP()); // Print the IP address

  server.on("/", handleRoot);
  server.on("/ledcontrol", Led_Control);
  server.begin();
  Serial.println("HTTP Server started");
}

```

Ilustración 4.23 Sketch Arduino ESP8266

En la siguiente Ilustración 4.24, se da de alta un servicio llamado Led\_Control, con la finalidad de interactuar con la iluminación dentro de la red, la sintaxis para encender la bombilla sería la siguiente <http://192.168.1.68/ledcontrol?ledstate=on>, la dirección ip puede variar según la configuración previa, en este caso el dispositivo, al conectarse a la red, adquiere una dirección aleatoria o dinámica, la configuración y resultado final se puede observar en la Ilustración 4.25.

```

void Led_Control()
{
  String message;
  bool state;
  String ledstate = server.arg("ledstate");
  if(ledstate == "on")
  {
    state = true;
    message = "LED ON";
  }
  else if(ledstate == "off")
  {
    state = false;
    message = "LED OFF";
  }
  else
  {
    state = false;
    message = "NO OPTION INDICATED";
  }

  digitalWrite(relay, state);
  Serial.println(message);
  server.send(200, "text/plain", message);
}

```

Ilustración 4.24 Sketch On & Off

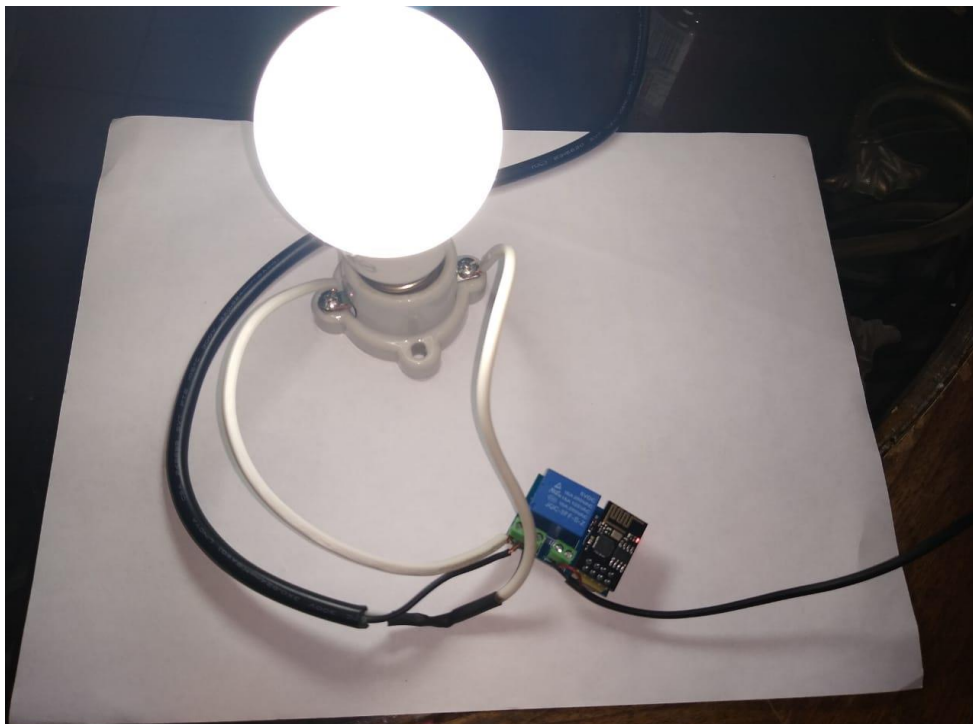


Ilustración 4.25 Modulo Esp8266

## 4.4 Envío de mensajería instantánea

En la Ilustración 4.26 se puede observar la interfaz para el envío de mensajería ya sea por WhatsApp o servicio de mensajes cortos haciendo uso de las mismas librerías, mediante un control de selección se escoge el contacto destino y se puede escribir desde el teclado de Windows y el SDK de Tobii el mensaje a enviar por parte del usuario.

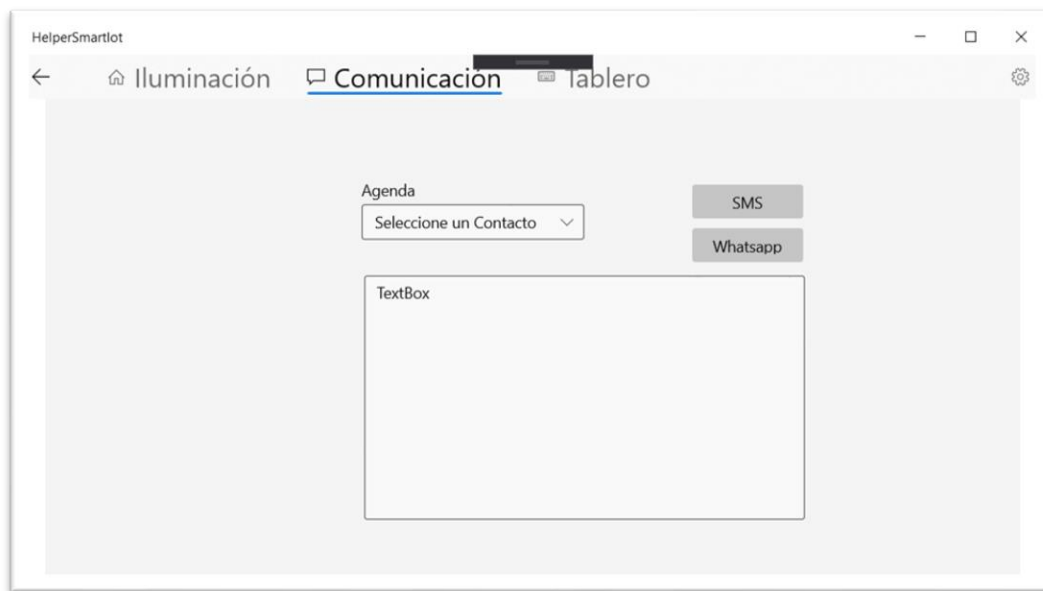


Ilustración 4.26 Interfaz Modulo Comunicación

Como se puede observar en la Ilustración 4.26 los controles que se incorporaron en el módulo de comunicación, es un combobox desplegable en el cual se desplegará el listado de contactos dados de alta en el sistema, después se incorporó un textbox, en este elemento se escribirá el mensaje por parte del usuario que se requiera enviar, utilizando la herramienta del teclado que provee la aplicación de Tobii tracker habilitando en Windows la accesibilidad para seguimiento ocular, los botones de SMS y Whatsapp, son para elegir el medio por el cual se pretende realizar el envío del mensaje.

## 4.5 Sintetizador voz por teclado

A continuación, en la Ilustración 4.27 se muestra la interfaz por la cual, a través del sistema, el usuario indica palabras claves a reproducir por los parlantes o bocinas, donde se encuentre instalado el sistema.

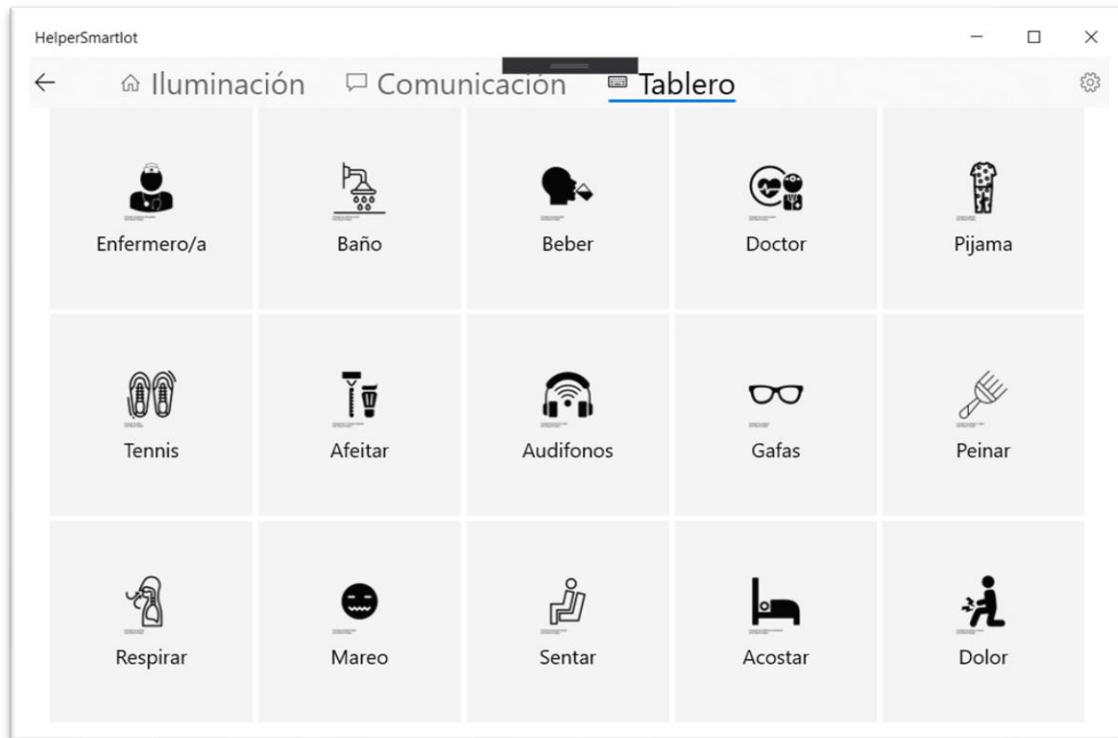


Ilustración 4.27 Interfaz Tablero



En la Ilustración 4.28 se utiliza un método para interpretar la petición, la voz se puede configurar desde Windows, adaptando a las preferencias de usuario, en este caso el paciente solicita un doctor.

```
#region Comunicacion Parlantes
0 referencias
private async void btnComunicacion_Click(object sender, RoutedEventArgs e)
{
    MediaElement mediaElement = new MediaElement();
    var synth = new Windows.Media.SpeechSynthesis.SpeechSynthesizer();
    Windows.Media.SpeechSynthesis.SpeechSynthesisStream stream =
    await synth.SynthesizeTextToStreamAsync("Necesito un doctor");
    mediaElement.SetSource(stream, stream.ContentType);
    mediaElement.Play();
}
#endregion
```

Ilustración 4.28 Comunicación mediante parlantes en C# .net

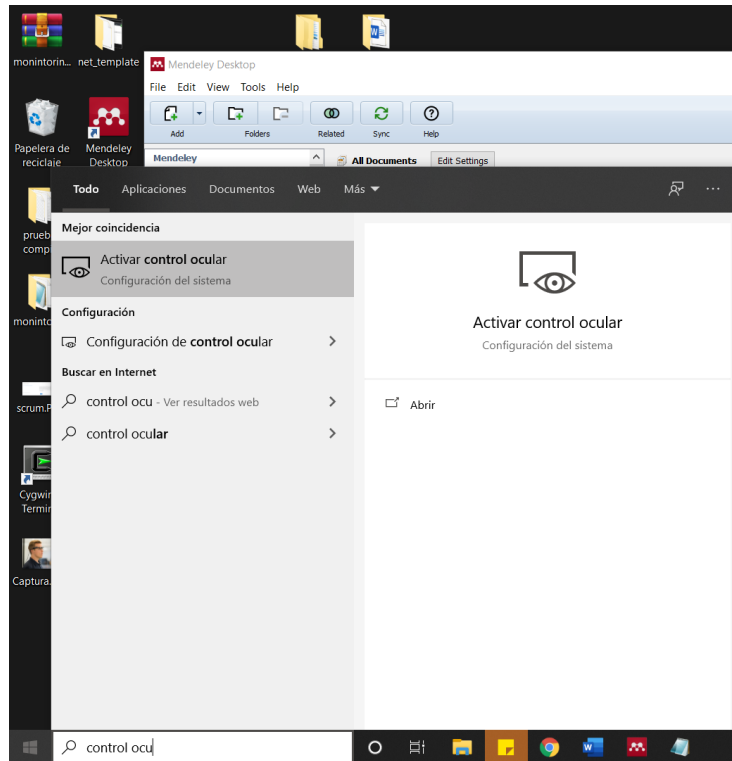
## 5 Prueba y Análisis de resultados

En este capítulo se realizará una reseña del análisis de resultados generados a partir de las pruebas realizadas por desarrolladores de software y usuarios promedio al sistema.

No se pudieron generar pruebas de inclusión a personas con ELA, debido a la emergencia sanitaria COVID-19, que se presentó justo en el último capítulo de la tesis.

### 5.1 Configuraciones Iniciales

Como parte del análisis de resultados se detectaron algunas configuraciones que se pueden realizar para mejorar la experiencia de usuario en el sistema. Se recomienda habilitar en el sistema operativo de Windows accesibilidad seguimiento de mirada, esto se hace escribiendo control ocular en el inicio de Windows, posteriormente se selecciona la opción, como se muestra en la barra de inicio en la Ilustración 5.1 .



**Ilustración 5.1 Control ocular inicio Windows**

También es recomendable activar la tecnología de Windows Hello, la cual mejora la experiencia de usuario pues da soporte al reconocimiento facial con seguridad biométrica, por lo tanto, al encender o apagar el computador la persona puede acceder al sistema sin necesidad de escribir en el teclado físicamente las credenciales de Windows, estas dos configuraciones descritas aportan la posibilidad de lanzar el ejecutable del sistema domótico, con la mirada.

## 5.2 Modo Nocturno

El modo nocturno o modo oscuro es una adaptación que se ofrecen en muchos dispositivos digitales, para reducir el brillo de la pantalla y disminuir el esfuerzo ocular, en lugar de presentar un fondo con colores blancos y llamativas letras coloridas, el modelo oscuro típico presenta un fondo negro con letras blancas.

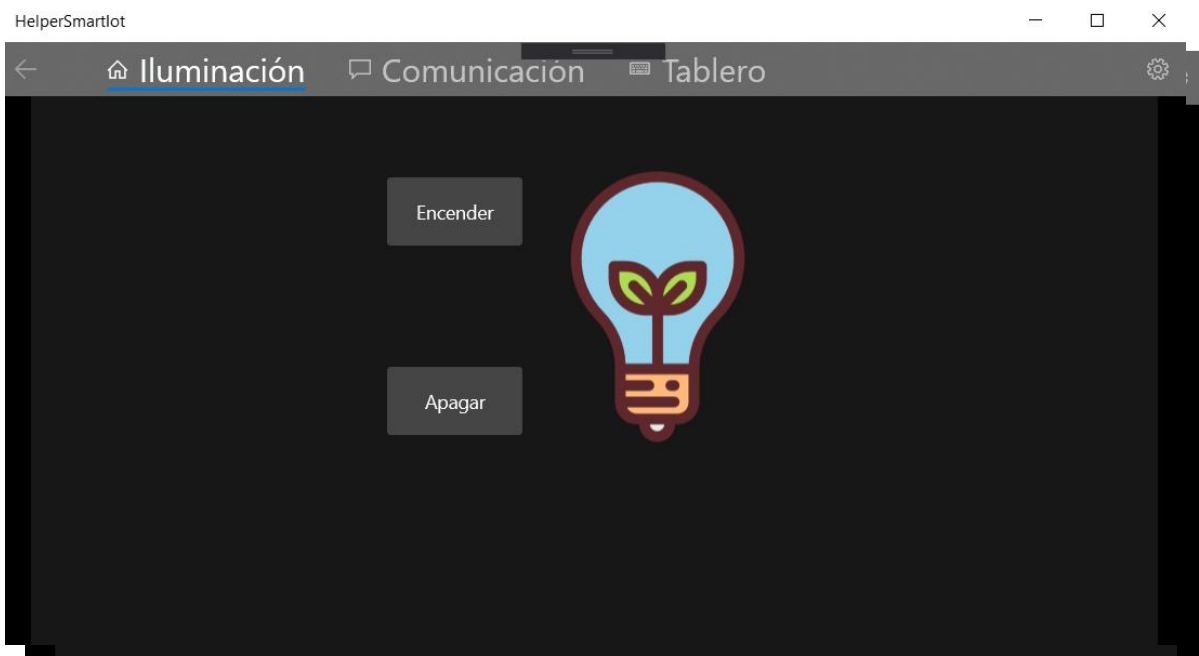
La combinación que se utiliza en el modo nocturno reduce el destello y

ayudan a que nuestros ojos se adapten mejor a la luz, lo que ayuda a reducir el esfuerzo y poder pasar más tiempo sin cansarnos frente a la pantalla de los dispositivos.

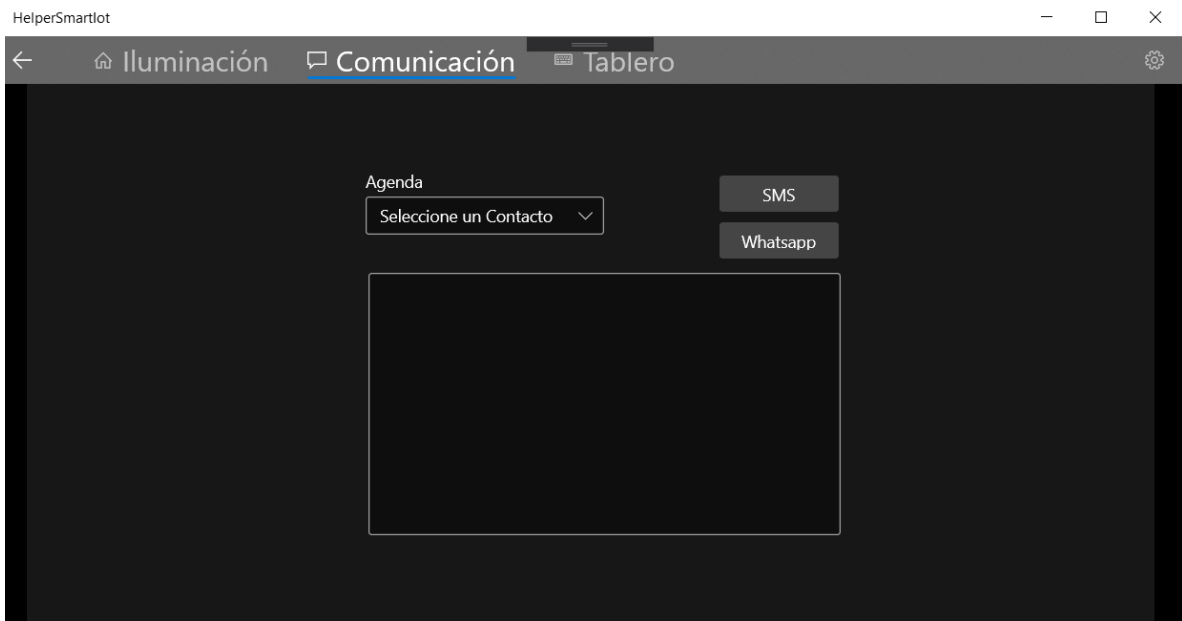


**Ilustración 5.2 Registro inicio sesión biométrico**

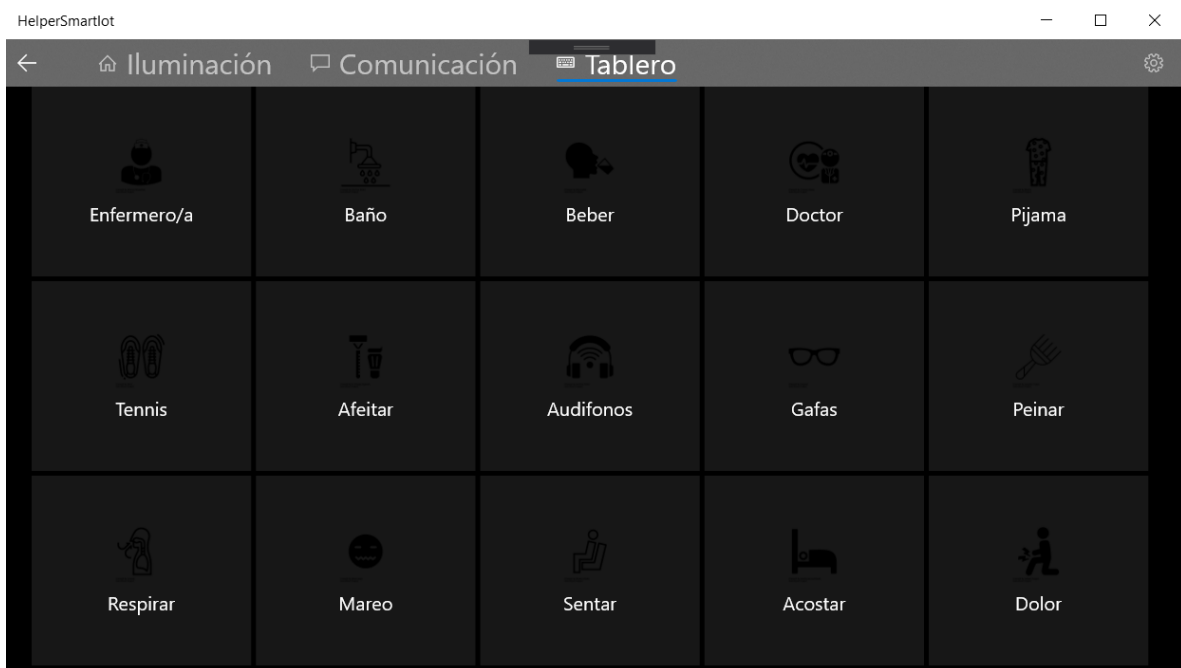
Es importante aportar desarrollos de software inclusivos que den bienestar mediante tecnologías amigables a la salud, las personas con ELA conservan los músculos de los ojos en buen estado, por lo tanto, se optó por tener en disponibilidad esta opción en el sistema, como se muestra en las Ilustraciones 5.3, 5.4, 5.5 y 5.6 de los diferentes módulos.



**Ilustración 5.4 Módulo Iluminación encendida**



**Ilustración 5.5 Módulo comunicación**



**Ilustración 5.6 Módulo tablero**

## **5.3 Módulo Domótico Encendido Luz**

La arquitectura de triangulación en la comunicación tableta, Raspberry y el dispositivo IOT, se realizó con el objetivo de poder combinar diferentes módulos de comunicación mediante las librerías potentes que encapsula Python para hacer uso de embebidos, también el delegar la responsabilidad a una web API genera la oportunidad de poder hacer uso de los dispositivos conectados a la red desde diferentes plataformas, como relojes inteligentes, aplicaciones de celular, pantallas de carro, wereables, etc.

Sin embargo, esto también tiene sus desventajas, se necesita fortalecer la seguridad mediante filosofías de tokenización, la seguridad de los dispositivos del hogar inteligente es algo esencial en el internet de las cosas.

## **5.4 Módulo Comunicación**

En este módulo como se pudo observar de acuerdo a las pruebas, que es necesario crear componentes más grandes como lo es el seleccionado de contactos para el envío de mensajería instantánea, también consumir las webs API'S externas, para realizar el cargado dinámico de los contactos en el control combo box.

## **5.5 Módulo Tablero**

Una vez realizada las pruebas del tablero, se genera un diccionario en la base de datos con las palabras claves y con esto realizar un cargado dinámico en el tablero, para facilitar la integración de nuevas palabras vinculadas a los botones de acción, así, mediante la suscripción al mismo evento según la nomenclatura, se utilizará el mismo bloque de código y con esto sustentar las buenas prácticas de desarrollo.

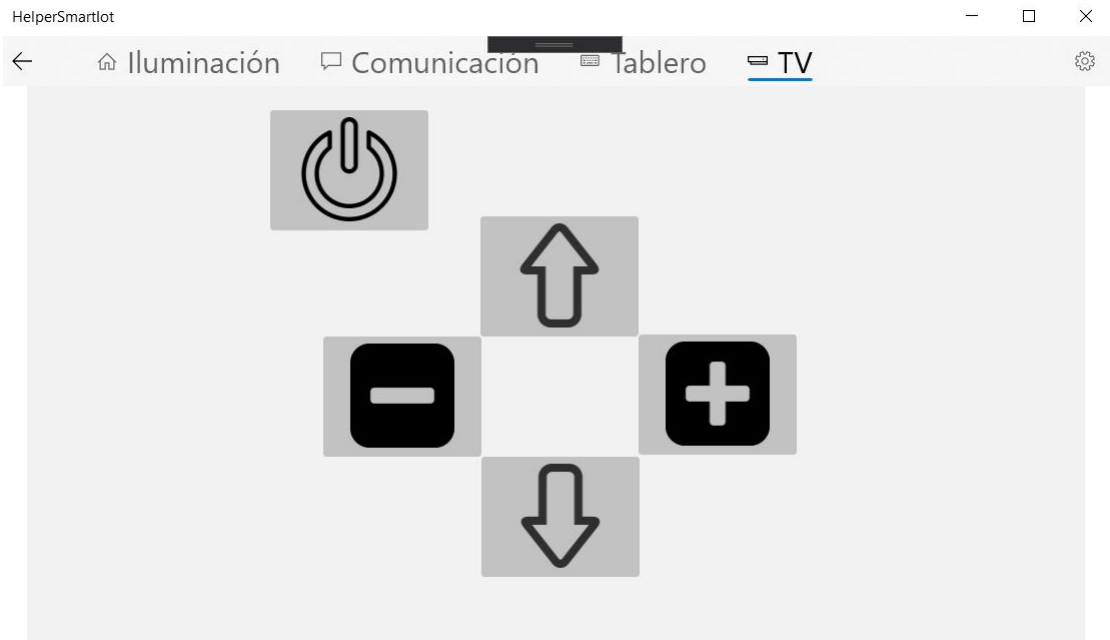
## 5.6 Pruebas Televisión

Un mando a distancia es usado para controlar el televisor como se muestra en la Ilustración 5.7 Control Televisor, usualmente este tipo de enlace de comunicación se establece por medio de señales infrarrojas, este mismo tipo de prácticas es reproducible para controles de sistemas de audio, refrigeración ambiental, etc.



Ilustración 5.7 Control Televisor

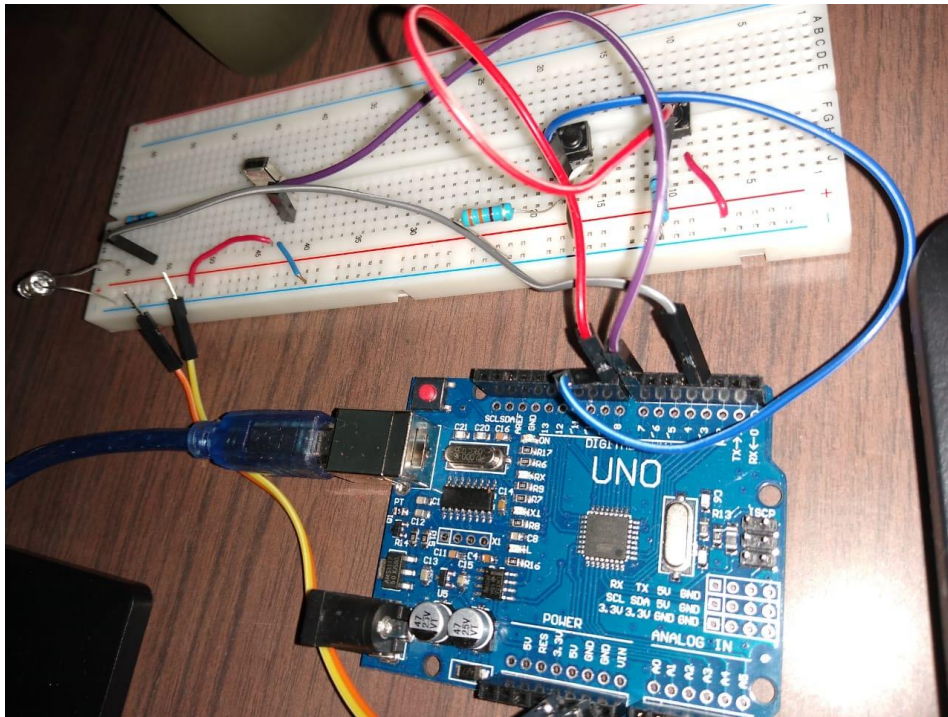
Se incorporó un módulo extra al sistema llamado tv, con la finalidad de ampliar el aspecto de domótica de la aplicación, con el cual se puede controlar la televisión de la habitación, tal y como se muestra en la interfaz gráfica de la Ilustración 5.8 Módulo TV, por el momento sólo se puede encender, apagar, subir/bajar volumen y cambiar de canales. El control tiene las mismas limitantes que un control convencional, necesita tener el área despejada del módulo emisor-receptor para llevar a cabo su funcionamiento.



**Ilustración 5.8 Módulo TV**



Se condicionó una tablilla de pruebas (protoboard) con leds infrarrojos receptores y emisores, y con ayuda de un prototipado Arduino se programó en C++ el código necesario para poder llevar a cabo la recolección de los códigos de los botones originales del control, para posteriormente lograr replicar la funcionalidad como se muestra en la Ilustración 5.9.



**Ilustración 5.9 Configuración Televisor**

Como se puede observar en la imagen anterior, el led emisor de control se encuentra apuntando hacia enfrente, emite las señales hacia el televisor directamente, el módulo receptor que se observa se utilizó para capturar directamente las señales del control a replicar, primero se extraen los códigos y se observan en la consola del IDE de Arduino, para posteriormente programarse en el control réplica.

Los botones en el protoboard son botones de funcionalidad manual, para generar las órdenes directamente desde la placa de ensayo, este tipo de control funcionó para un televisor que se muestra en la Ilustración 5.10.



Ilustración 5.10 TV LG

Se trató de replicar la funcionalidad para un quinto módulo para controlar el clima en la habitación, las pruebas se hicieron en dos equipos diferentes un modelo EXF121D marca MIRAGE y en el modelo SP122CM marca LG.

Sin embargo, no se pudieron recolectar los códigos por lo que parece ser un problema de compatibilidad del módulo receptor que se utilizó en la placa de ensayos.

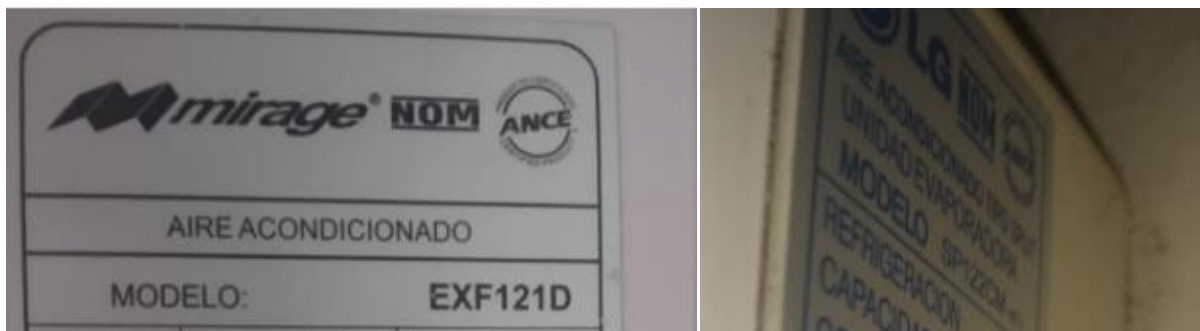


Ilustración 5.11 Modelos Minisplit

## 5.7 Validación del Sistema

Como previamente se comentó, no se pudieron generar pruebas de inclusión a personas con ELA debido a la emergencia sanitaria COVID-19, lo cual debe permitir la validación del sistema, además, se considera que esto es posible con la aplicación de una encuesta de calidad, este cuestionario es un método de inspección por parte de evaluadores expertos, para analizar la usabilidad del sistema final, de tal manera que se propone el siguiente cuestionario que está basado en el Criterio de Usabilidad **ISO/IEC25010**.

Las evaluaciones por inspección denominado Evaluación Heurística, es un método de valoración, donde sus principales exponentes son Molich & Nielsen [33]; y que consiste en analizar la aprobación de la interfaz gráfica del usuario, en base a unos principios de usabilidad mediante la observación de varios profesionales expertos en el tema.

El cuestionario tendrá la estructura (preguntas checklist y tres respuestas: si cumple, no cumple y no aplica) y la ejecución de éste serán por parte de expertos en el tema de usabilidad de software.

- **Si Cumple:** indica que no existe un problema
- **No Cumple:** indica que se presentan falta de usabilidad
- **No Aplica:** indica que el software no hace uso de ese principio.

### *Evaluación heurística del sistema para el control domótico como Soporte a Personas con Esclerosis Lateral Amiotrófica (ELA)*

INTELIGIBILIDAD	Si Cumple	No cumple	No aplica
¿La aplicación que está evaluando le permite cumplir con el objetivo de ésta?			
¿Los objetivos de la aplicación son claros?			
¿La aplicación que está evaluando, le ofrece la información que usted necesita para realizar la acción deseada?			
¿La aplicación que está evaluando, se adapta a las necesidades, en cuanto a los procesos que usted como usuario necesita?			
APRENDIZAJE			
¿La aplicación le ofrece ventanas y enlaces claros?			
¿La aplicación que usted está evaluando, es manejable y le ofrece las ayudas necesarias para entenderla?			
¿La aplicación obliga a memorizar datos, procesos o imágenes para utilizarla?			
¿Los botones e imágenes que le ofrece la aplicación son fácil de entender?			
OPERABILIDAD			
¿Es Familiar el software con otras interfaces gráficas?			
¿Se encuentran Guías de la aplicación para un mejor uso?			
¿En el aplicativo se presentan imágenes que permitan entender el contenido?			

¿El aplicativo puede manejarse en los diferentes dispositivos como son celulares, Tablet y computadoras?			
¿El aplicativo puede manejarse y observarse desde cualquier navegador?			
<b>PROTECCIÓN</b>			
¿Existen mensajes de advertencia de errores dentro de la aplicación?			
¿Se guía al usuario a entender la información que se requiere en las cajas de textos o formatos que se presentan?			
¿El aplicativo le ofrece ayudas o guías para entender su entorno y las funciones que posee?			
¿El aplicativo presenta mensajes de advertencia antes de culminar un proceso?			
<b>ESTÉTICA</b>			
¿La interfaz cumple con los colores adecuados y coherentes en todo el aplicativo?			
¿El tipo de letra que se maneja en el aplicativo está dentro de las sans serif (verdana, arial, helvética)?			
¿Le agrada la interfaz que maneja el aplicativo?			
¿Es fácil de entender cómo funciona el aplicativo y las ayudas que le ofrece a través de la interfaz gráfica?			
¿El aplicativo le ofrece medios para comunicarse con el sistema y poder exponer sus recomendaciones o fallas?			
<b>ACCESIBILIDAD</b>			
¿Las imágenes tienen un tamaño indicado?			
¿El color del texto contrasta con la interfaz?			
¿Se puede acceder a la información a través de un lector de pantalla?			
¿Los sonidos que tiene el aplicativo son molestos?			
¿El aplicativo tiene música de fondo?			
¿La música de fondo del aplicativo es molesta?			
<b>Observaciones:</b>			

# 6 Conclusiones y trabajos futuros

A partir del trabajo desarrollado, el prototipo y el análisis de resultados se pretende ayudar a las personas con problemas de movilidad a mejorar su calidad de vida, fomentando la independencia de las personas en relación a su entorno.

## 6.1 Conclusiones

La presente investigación nos permite concluir que el sistema de domótica a través del seguimiento de mirada descrito ofrece a las personas con problemas de movilidad la capacidad de vencer las barreras de las limitantes arquitectónicas en el hogar y la capacidad de comunicación, este tipo de tecnologías inclusivas ayuda a las personas empoderándolas con habilidades de autonomía sobre su entorno.

Los casos de éxito identificados en la investigación fue la incorporación de un dispositivo de seguimiento de mirada para poder comunicarse y controlar las cosas en el hogar, el sistema es multiplataforma, se puede utilizar en laptops, desktops y tabletas.

Debido al paradigma de triangulación de comunicación de tecnologías, el hogar inteligente se puede gestionar mediante relojes inteligentes, celulares, sistemas web y diversos sistemas operativos.

El módulo de comunicación utilizado en esta investigación fue el esp8266, el cual es menos potente que un xbee o esp32, las diferencias radican en los tipos de tecnologías que pueden llegar a tener, el espe8266 es menos potente que los dos mencionados, debido al procesamiento, entre otras características, sin embargo, el costo es mucho menor.

## 6.2 Trabajos Futuros

Es importante considerar como trabajo futuro el realizar la encuesta de calidad que se describió en el capítulo anterior como validación del sistema.

Asimismo, a continuación, se presentan algunas ideas y mejoras para el sistema como lo son el consumo energético, la impresión aditiva 3D, la

adaptación de sistemas embebidos en PCB'S y el sistema desde una perspectiva de monitoreo por parte de especialistas de la salud.

**Consumo Energético:** La domótica y el seguimiento de mirada abren muchos caminos hacia la investigación, a trabajos futuros para mejorar los siguientes aspectos: Cada dispositivo del hogar a controlar debería tener embebido en su sistema un control de consumo eléctrico, para gestionar el gasto y contribuir con el medio ambiente.

**Impresión 3D:** El dispositivo de seguimiento de mirada se adapta mediante una cinta adhesiva imantada en el monitor, lo cual lo vuelve de cierta manera vulnerable a caídas cuando se somete a la interacción, por lo tanto, se propone el hacer un dispositivo adaptador como el que se muestra en la Ilustración 6.1.



Ilustración 6.1 Display Acoplador, recuperada de [33].

Hoy en día la fabricación aditiva o impresión 3D, hace posible el diseño de bricolaje de cualquier tipo, para adaptar los dispositivos para hacer productos a la medida del usuario final y dejando como límite la imaginación del desarrollador, facilitando la adaptación de cualquier componente sin necesidad de salir de casa.

El costo de una impresora 3D es relativamente económico a medida que avanza la tecnología, cada vez salen al mercado impresoras más económicas. Con el software necesario se pueden imprimir modelos rápidamente; además, en internet se pueden encontrar algunas plantillas predefinidas para descargar e imprimir, por otra parte, también hay servicios en línea de impresión, que al subir tu modelo hacen llegar el producto final hasta la puerta de tu casa.

**Adaptación de Embebidos:** La incorporación de una computadora de bajo costo como web API y como coordinador del hogar automatizado, requiere de un diseño decente como un producto final, hay algunas páginas con información de cómo adaptar un PCB con la finalidad de ser lo más minimalista posible y presentable al conectarse en el suministro eléctrico del hogar volviendo el componente más atractivo físicamente, en la Ilustración 6.2 se puede observar una adaptación para un Raspberry zero .

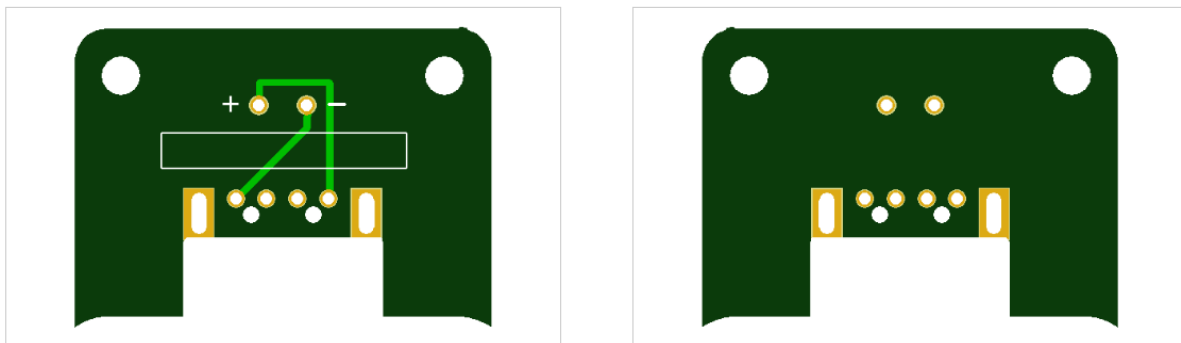


Ilustración 6.2 Pcb raspberry zero

El proceso de creación de PCB's a la medida, tiene como objetivo realizar adaptaciones que se vean presentables de manera que puedan ser introducidas en las cosas de nuestro hogar y que pase desapercibido sus dimensiones, hoy en día existen muchos servicios en línea, sin embargo, China te ofrece los paquetes más económicos.

Como trabajo futuro se considera crear un PCB por cosa del hogar que se desea incorporar al sistema, adaptar una base impresa 3d donde esté embebido el sistema, según sea el caso y cumplir con las normativas ISO, para no poner en riesgo la integridad del hogar.

**Soporte de especialistas de la salud:** Las personas con ELA, como se mencionó en capítulos anteriores, usualmente necesitan de diferentes tipos de terapia, sería de gran ayuda incorporar este tipo de tecnologías a una red de profesionales de la salud para una atención vía web API'S, para fortalecer una relación paciente especialista donde pudieran consultar por mensajería instantánea, comunicarse directamente por medio de la tableta.

Otro aspecto de gran impacto es el monitoreo de la salud, las personas con ELA usualmente fallecen por problemas respiratorios, la incorporación de un sensor de sonido como lo es un micrófono o un estetoscopio digital como el de la Ilustración 6.1 Ilustración 6.1 Estetoscopio Digital, sería de gran utilidad para llevar a cabo el análisis de la respiración y los sonidos pulmonares del paciente, con la finalidad de predecir un paro respiratorio a medida del progreso de la enfermedad, con ayuda de técnicas de aprendizaje profundo [34].

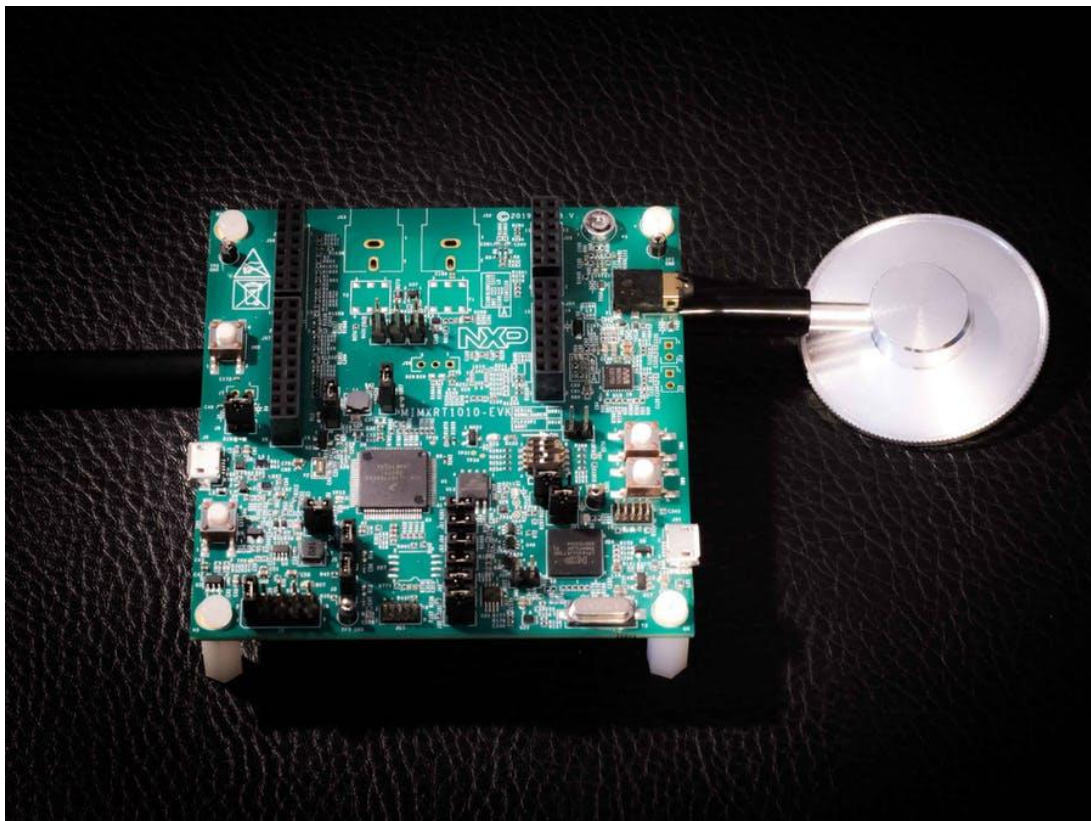


Ilustración 6.1 Estetoscopio Digital



# Bibliografía

- [1] C. N. para el D. y la I. de las P. con Discapacidad, “La Esclerosis Lateral Amiotrófica ELA | Consejo Nacional para el Desarrollo y la Inclusión de las Personas con Discapacidad | Gobierno | gob.mx.” <https://www.gob.mx/conadis/es/articulos/la-esclerosis-lateral-amiotrofica-ela?idiom=es>. (accessed Mar. 04, 2020).
- [2] A. I. G. A.C., “¿Qué es la ELA? - Gila - Apoyo integral para pacientes con ELA.” <https://www.ela.org.mx/la-ela/> (accessed Mar. 04, 2020).
- [3] Expansion, “Esclerosis Lateral Amiotrófica, una enfermedad muy costosa.” <https://expansion.mx/mi-dinero/2014/08/27/cuanto-cuesta-vivir-con-ela> (accessed Mar. 04, 2020).
- [4] Teleton, “Terapia Ocupacional, una puerta a la libertad - Teletón México.” <https://teleton.org/terapia-ocupacional-una-puerta-a-la-libertad/> (accessed Mar. 04, 2020).
- [5] A. P. Rose Grijalva, “Diseño de espacio a través de la intervención tecnológica para personas con capacidades especiales,” CEDIM, Monterrey, México., 2010.
- [6] O. A. Grijalva Hernández, “SISTEMA DE MONITORIZACIÓN OCULAR QUE PERMITA LA COMUNICACIÓN CON PACIENTES CUADRIPLÉJICOS,” Instituto Tecnológico de Hermosillo.
- [7] Ultimahora, “Tecnología permite a paciente con esclerosis controlar ordenador con la vista | Guatemala, Inteligencia artificial, Microsoft, Personas con discapacidad, Stephen Hawking, WhatsApp.” <https://www.ultimahora.com/tecnologia-permite-paciente-esclerosis-controlar-ordenador-la-vista-n2788410.html> (accessed Mar. 20, 2020).
- [8] A. R. T. y “El T. de C. Tlacuany”, “Andres - Tablero - Plataforma Afectados de ELA-Compendio de Información.” <https://sites.google.com/site/plataformadeafectadosela/etran/clic/andres> (accessed Mar. 20, 2020).
- [9] Escrito por el personal de Mayo Clinic, “Esclerosis lateral amiotrófica,” 2019. <https://www.mayoclinic.org/es-es/diseases-conditions/amyotrophic-lateral-sclerosis/symptoms-causes/syc-20354022>.
- [10] A. K. Dennis, *Raspberry Pi Home Automation with Arduino Second Edition Unleash the power of the most popular microboards to build convenient, useful, and fun home automation projects*. 2015.
- [11] M. Schwartz et al., *Internet of Things with the Arduino Yún Projects to help you build a world of smarter things Internet of Things with the Arduino Yún Credits Commissioning Editor*. 2014.
- [12] P. Mulder and K. Breseman, “Node.js for Embedded Systems BUILDING WEB INTERFACES FOR CONNECTED DEVICES.” Accessed: Mar. 20, 2020. [Online]. Available: [www.allitebooks.com](http://www.allitebooks.com).
- [13] W. Bolton, *Mecánica sistemas de control electrónico en la ingeniería mecánica y eléctrica*. Edinburgh Gate, Inglaterra, 2008.
- [14] M. B. H. David G. Alciatore, *Introducción a la Mecatrónica y los sistemas de medición*, 3rd ed. (338). México DF, 2008.
- [15] M. Schwartz, M. Schwartz Reviewer, and C. Batrinu, *Internet of Things with ESP8266 Build amazing Internet of Things projects using the ESP8266 Wi-Fi chip*

*Internet of Things with ESP8266 Credits*. 2016.

- [16] “DEVKITV1 ESP32 de 30 Pines con Wifi y Bluetooth.”  
<https://www.cdmxelectronica.com/producto/devkitv1-esp32-modulo-wifi-bluetooth-esp32-arduino/> (accessed Jun. 16, 2020).
- [17] “How to Install Full Windows 10 on a Raspberry Pi | Tom’s Hardware.”  
<https://www.tomshardware.com/reviews/install-windows-10-on-raspberry-pi,5993.html> (accessed Jun. 17, 2020).
- [18] Eyetrackinsights, “Eye Insights: Eye Tracking y Neuromarketing en México - Alianzas - Agencias.” <https://www.eyetrackinsights.com/> (accessed Mar. 05, 2020).
- [19] tobiidynavox, “¿Cómo funciona el seguimiento ocular? - Tobii Dynavox.”  
[https://www.tobiidynavox.com/en-US/about/about-us/how-eye-tracking-works/?MarketPopupClicked=true&utm\\_medium=organic&utm\\_source=www.google.com](https://www.tobiidynavox.com/en-US/about/about-us/how-eye-tracking-works/?MarketPopupClicked=true&utm_medium=organic&utm_source=www.google.com) (accessed Mar. 05, 2020).
- [20] Irisbond, “Comunicación aumentativa y alternativa | Irisbond.”  
<https://www.irisbond.com/comunicacion-aumentativa-alternativa/> (accessed Mar. 20, 2020).
- [21] “TM5 Mini Eye Tracker - EyeTech Digital Systems.” <https://eyetechds.com/eye-tracking-products/tm5-mini-eye-tracker/> (accessed Mar. 20, 2020).
- [22] “SMI RED-m - iMotions.” <https://imotions.com/hardware/smi-red-m/> (accessed Mar. 20, 2020).
- [23] “Tobii Eye Tracker 4C para juegos de PC. Compre ahora a € 169 - Tobii Gaming.”  
<https://gaming.tobii.com/tobii-eye-tracker-4c/> (accessed Mar. 20, 2020).
- [24] A. Papoutsaki, J. Laskey, A. Gokaslan, Y. He, and J. Huang, “WebGazer.js - Democratizing Webcam Eye Tracking on the Browser,” 2018.  
<https://webgazer.cs.brown.edu/> (accessed Jun. 16, 2020).
- [25] “Twilio alternative.” [https://www.bitrix24.com/alternatives/free-twilio-alternative.php?gclid=Cj0KCQjw9tbzBRDVARIsAMBplx9OybMcDPQN63DhwpGvVSEcdqcEY3xMicSeuokZG9yA0sSoFD0M\\_6UaAkzBEALw\\_wcB](https://www.bitrix24.com/alternatives/free-twilio-alternative.php?gclid=Cj0KCQjw9tbzBRDVARIsAMBplx9OybMcDPQN63DhwpGvVSEcdqcEY3xMicSeuokZG9yA0sSoFD0M_6UaAkzBEALw_wcB) (accessed Mar. 21, 2020).
- [26] “Enviar mensaje de Whatsapp - NiceApi.Net.” <https://niceapi.net/> (accessed Mar. 21, 2020).
- [27] Wyze, “Smart Home Starter Bundle Includes Camera, Contact Sensor (2), Motion Sensor, Bulb (3), Plug (2), SD Card,” 2019. .
- [28] Google, “Nest Thermostat E,” 2019. <https://www.homedepot.com/p/Google-Nest-Thermostat-E-T4000ES/302714540>.
- [29] google, “Nest-Hub-Chalk,” 2019. <https://www.homedepot.com/p/Google-Nest-Hub-Chalk-GA00516-US/306809073>.
- [30] Amazon, “Presentamos Echo Studio - Bocina inteligente de alta fidelidad con Alexa,” 2019, [Online]. Available:  
[https://www.amazon.com.mx/gp/product/B07NQHB1D8/ref=s9\\_acsd\\_al\\_bw\\_c\\_x\\_1\\_w?pf\\_rd\\_m=A3TO6F13CSVUA4&pf\\_rd\\_s=merchandised-search-20&pf\\_rd\\_r=06HMNZBYF6FPEA9080R5&pf\\_rd\\_t=101&pf\\_rd\\_p=0e7af64f-6aed-4cc3-812d-528587331d05&pf\\_rd\\_i=19091376011](https://www.amazon.com.mx/gp/product/B07NQHB1D8/ref=s9_acsd_al_bw_c_x_1_w?pf_rd_m=A3TO6F13CSVUA4&pf_rd_s=merchandised-search-20&pf_rd_r=06HMNZBYF6FPEA9080R5&pf_rd_t=101&pf_rd_p=0e7af64f-6aed-4cc3-812d-528587331d05&pf_rd_i=19091376011).
- [31] Apple, “El nuevo sonido de la casa.,” 2019, [Online]. Available:  
[https://www.apple.com/mx/homepod/?afid=p238%7CsuILlcy5z-dc\\_mtid\\_20925su964010\\_pcrd\\_306102590073\\_&cid=wwa-mx-kwgo-aes-slid-](https://www.apple.com/mx/homepod/?afid=p238%7CsuILlcy5z-dc_mtid_20925su964010_pcrd_306102590073_&cid=wwa-mx-kwgo-aes-slid-)
- [32] A. Bissoli, D. Lavino-Junior, M. Sime, L. Encarnação, and T. Bastos-Filho, “A

human–machine interface based on eye tracking for controlling and monitoring a smart home using the internet of things,” *Sensors (Switzerland)*, vol. 19, no. 4, pp. 1–26, 2019, doi: 10.3390/s19040859.

- [33] N. Sánchez León, M. Rivera Guzmán, A. Moreno Vargas, and M. Díaz Molina, “Heuristic evaluations: App movil para evaluaciones heurísticas de la usabilidad e ISO25010,” *Maskana*, vol. 7, no. Supl., pp. 1–11, 2017.
- [34] C. Jácome, J. Ravn, E. Holsbø, J. C. Aviles-Solis, H. Melbye, and L. A. Bongo, “Convolutional neural network for breathing phase detection in lung sounds,” *Sensors (Switzerland)*, vol. 19, no. 8, Apr. 2019, doi: 10.3390/s19081798.