



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E**  
**INDUSTRIAL**  
**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES**

**TEMA:**

---

“SISTEMA ELECTRÓNICO PARA FACILITAR LA COMUNICACIÓN Y  
ACTIVIDADES COTIDIANAS DE LOS ENFERMOS CON ESCLEROSIS  
LATERAL AMIOTRÓFICA (ELA)”

---

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Electrónica y Comunicaciones.

**LÍNEAS DE INVESTIGACIÓN:** Sistemas de Control

**AUTOR:** Collaguazo Chicaiza Jorge Héctor

**TUTOR:** Ing. Córdova Cordova Edgar Patricio, Mg.

**Ambato – Ecuador**

**Febrero, 2018**

## **APROBACIÓN DEL TUTOR**

En mi calidad de Tutor del Trabajo de Investigación sobre el Tema:

“SISTEMA ELECTRÓNICO PARA FACILITAR LA COMUNICACIÓN Y ACTIVIDADES COTIDIANAS DE LOS ENFERMOS CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)”, del señor. Collaguazo Chicaiza Jorge Héctor, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los tramites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, febrero del 2018

**EL TUTOR**



---

**Ing. Patricio Córdova, Mg.**

## **AUTORÍA**

El presente Proyecto de Investigación titulado: “SISTEMA ELECTRÓNICO PARA FACILITAR LA COMUNICACIÓN Y ACTIVIDADES COTIDIANAS DE LOS ENFERMOS CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)”, es absolutamente original, autentico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, febrero del 2018



---

**Collaguazo Chicaiza Jorge Héctor**

**CC: 1805086228**

## **DERECHOS DE AUTOR**

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, febrero del 2018



---

**Collaguazo Chicaiza Jorge Héctor**

**CC: 1805086228**

## APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Manuel Otorongo, e Ing. Patricio Encalada, reviso y aprobó el Informe Final del Proyecto de Investigación titulado “SISTEMA ELECTRÓNICO PARA FACILITAR LA COMUNICACIÓN Y ACTIVIDADES COTIDIANAS DE LOS ENFERMOS CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)”, presentado por el señor Collaguazo Chicaiza Jorge Héctor de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.



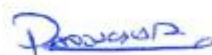
Ing. Mg. Elsa Pilar Urrutia Urrutia

PRESIDENTA DEL TRIBUNAL



DOCENTE CALIFICADOR

Ing. Mg. Manuel Otorongo



DOCENTE CALIFICADOR

Ing. Mg. Patricio Encalada

## **DEDICATORIA**

*A Dios, por haberme regalado la vida, por llevarme siempre en sus brazos y protegerme, por fortalecer mi corazón e iluminar mi mente, por darme salud y sabiduría para poder llegar a este momento tan especial de mi vida profesional.*

*A mi madre María Chicaiza, por ser el pilar fundamental de mi vida, por sus consejos, sus valores, por la motivación constante que me ha permitido ser una persona de bien, pero más que nada, por su esfuerzo y sacrificio incondicional durante todo el trayecto de mi vida universitaria.*  
*A mi padre Jorge, por los ejemplos de lucha y perseverancia demostrados para salir adelante.*

*A mis hermanos: Marcelo, Esther, Carlos y mi cuñado Raúl, quienes siempre me apoyaron con sus palabras de aliento, y me brindaron la confianza de contar con ellos en todo momento.*

*A mi sobrino Erick, para que veas en mi un ejemplo a seguir.*

***Collaguazo Chicaiza Jorge Héctor***

## **AGRADECIMIENTO**

*Agradezco infinitamente a Dios por darme la oportunidad y la dicha de superarme.*

*A mis padres, hermanos y familiares, por estar siempre al pendiente de mí, motivándome para cumplir mis objetivos.*

*A mi asesor de Tesis el Ing. Patricio Córdova, por su tiempo, por confiar en mi capacidad y sobre todo por guiarme durante el desarrollo de este proyecto de investigación.*

*A la Facultad de Ingeniería en Sistemas, Electrónica e Industrial y sus docentes por formarme como persona y buen profesional.*

*A mi enamorada Digna, quien durante los últimos años de mi carrera me ha apoyado para continuar y no renunciar, gracias por tu amor incondicional.*

*Finalmente, agradezco a todos mis amigos, en especial a Homero, Cristian y Walter por colaborar con ideas y buenos deseos para que este proyecto se lleve a cabo, gracias por demostrarme que siempre contare con ustedes.*

**Collaguazo Chicaiza Jorge Héctor**

## ÍNDICE

APROBACIÓN DEL TUTOR .....	ii
AUTORÍA .....	iii
DERECHOS DE AUTOR .....	iv
APROBACIÓN DE LA COMISIÓN CALIFICADORA .....	v
DEDICATORIA .....	vi
AGRADECIMIENTO .....	vii
ÍNDICE.....	viii
ÍNDICE DE FIGURAS .....	xiii
ÍNDICE DE TABLAS .....	xv
RESUMEN .....	xvi
ABSTRACT .....	xvii
GLOSARIO DE TÉRMINOS Y ACRÓNIMOS .....	xvii
INTRODUCCIÓN .....	xix
CAPÍTULO 1.....	1
EL PROBLEMA.....	1
1.1 Tema .....	1
1.2 Planteamiento del problema .....	1
1.3 Delimitación .....	2
1.4 Justificación .....	3
1.5 Objetivos.....	4
1.5.1 Objetivo General.....	4
1.5.2 Objetivos Específicos .....	4
CAPÍTULO 2.....	5
MARCO TEÓRICO .....	5
2.1 Antecedentes Investigativos .....	5
2.2 Fundamentación teórica.....	8
2.2.1 Esclerosis Lateral Amiotrófica (ELA).....	8
Causas de ELA .....	8
Diagnósticos de ELA .....	9
Síntomas iniciales de ELA.....	9
Diferentes etapas de ELA .....	10



Tratamientos de la enfermedad de ELA .....	10
2.2.2 Sistemas de Comunicación Aumentativa y Alternativa (SAAC).....	11
Estrategias para la Comunicación Aumentativa y Alternativa.....	11
a) La selección directa .....	11
b) La selección con ratón.....	12
c) La exploración o barrido.....	12
d) La exploración o barrido independiente.....	12
e) La selección codificada .....	12
2.2.3 Dispositivos Electrónicos de Comunicación para enfermos con ELA.....	12
Sistema ACAT.....	13
Sistema Eye-Traking .....	14
Tobii Dynavox .....	15
2.2.4 Sistema Electrónico .....	15
a) Dispositivos de Entrada.....	16
Sensor .....	16
Sensor de proximidad .....	16
Tipos de sensores de proximidad.....	16
Sensor infrarrojo .....	16
b) Dispositivos de Proceso.....	17
Raspberry Pi.....	17
BeagleBone.....	17
Jaguarboard.....	18
c) Dispositivos de Salida .....	18
Actuadores .....	18
2.2.5 Domótica.....	18
Aportes de la domótica .....	19
2.2.6 Software libre.....	20
Libertades esenciales .....	20
2.2.7 Comunicación inalámbrica .....	21
Tipos de sistemas de comunicación inalámbrica.....	21
Televisión y Radiodifusión.....	22
Comunicación por satélite .....	22
Sistema de teléfono móvil (comunicación celular) .....	22
Red de área local inalámbrica (WLAN) .....	22

Comunicación Bluetooth .....	23
2.3 Propuesta de solución .....	24
CAPÍTULO 3.....	25
METODOLOGÍA.....	25
3.1 Modalidad de la investigación.....	25
3.2 Población y muestra.....	25
3.3 Recolección de la información .....	26
3.4 Procesamiento y análisis de datos .....	26
3.5 Desarrollo del proyecto .....	26
CAPÍTULO 4.....	28
PROPUESTA .....	28
4. 1 Descripción general del Sistema.....	28
4.2 Análisis del movimiento de los párpados.....	29
Movimiento de los párpados en pacientes con ELA .....	29
4.3 Análisis Técnico de Hardware.....	29
4.3.1 Sistema de Sensorización .....	29
4.3.2 Tarjeta controladora central.....	31
4.3.3 Comparativa entre placas electrónicas microcontrolador.....	35
4.3.4 Comparativa entre pantallas Touch Screen .....	37
4.3.5 Parlantes.....	38
4.3.6 Comparación de Tecnología inalámbrica .....	39
4.4 Obtención de datos .....	41
4.4.1 Detector de Proximidad .....	41
Módulo del sensor TCRT5000 .....	41
Ubicación del Sensor en los lentes .....	42
Calibración del sensor.....	43
Diagrama eléctrico empleado en los lentes del paciente .....	43
Análisis de consumo energético .....	44
Implementación del prototipo para los lentes.....	47
4.4.2 Detección del Movimiento de Cabeza.....	48
Módulo de la cámara RPI .....	48
4.5 Análisis de Software .....	49
Comparación de los sistemas operativos .....	49
Distribuciones de Linux para Raspberry Pi .....	51

4.6 Diseño del sistema Propuesto .....	53
4.7 Diagrama general del sistema programado.....	54
4.8 Desarrollo del Sistema .....	56
4.8.1 Instalación de Ubuntu Mate en RPI 3 .....	56
4.8.2 Eviacam para usar el módulo de la cámara RPI .....	58
4.8.3 Instalación de librerías para Python.....	59
TKINTER .....	60
ESPEAK .....	61
PYBLUEZ .....	62
AUTOPY .....	64
4.8.4 SMPlayer para reproducir música .....	65
4.9 Interfaz gráfica para el paciente.....	66
4.10 Implementación de la interfaz gráfica para el paciente. ....	66
Análisis de Consumo Energético.....	67
Conexiones de dispositivos a la RPI 3.....	68
Prototipo final de la interfaz grafica .....	69
4.11 Implementación del sistema domótico .....	70
Diagrama eléctrico del sistema domótico.....	71
Prototipo final del sistema domótico .....	72
4.12 Pruebas de Funcionamiento.....	73
4.12.1 Prueba realizada con el sensor TCRT5000.....	73
Teclado Virtual .....	73
Funciones Rápidas (FR) .....	74
Active Control .....	75
Escuchar Música.....	76
4.12.2 Prueba realizada con la cámara RPI .....	77
4.13 Análisis de resultados .....	78
4.14 Análisis económico del proyecto.....	81
4.14.1 Costo de Hardware y Software .....	81
4.14.2 Costo inicial del diseño.....	83
4.14.3 Costo total del sistema.....	83
CAPÍTULO 5.....	85
CONCLUSIONES Y RECOMENDACIONES .....	85
5.1 Conclusiones.....	85

5.2 Recomendaciones .....	86
BIBLIOGRAFÍA .....	88
ANEXOS .....	94

## ÍNDICE DE FIGURAS

Figura 2.1: Esclerosis Lateral Amiotrófica (ELA). [11] .....	8
Figura 2.2: Sistema ACAT de Stephen Hawking. [20].....	14
Figura 2.3: Seguimiento ocular Eye-Traking. [22] .....	14
Figura 2.4: Dispositivo desarrollado por Tobii Dynavox. [24].....	15
Figura 2.5: Sistema Electrónico [26].....	15
Figura 2.6: Automatización domótica de una vivienda. [37].....	19
Figura 2.7: Aportes de la domótica .....	20
Figura 4.1: Organización jerárquica de actividades desarrolladas en el sistema .....	28
Figura 4.2: Parlantes para conectar a la placa RPI 3.....	39
Figura 4.3: Módulo del sensor TCRT5000 .....	41
Figura 4.4: Circuito Eléctrico del módulo TCRT5000 .....	42
Figura 4.5: Ubicación del sensor en los lentes .....	42
Figura 4.6: Diagrama eléctrico para la etapa de lentes .....	44
Figura 4.7: Prototipo final de los lentes .....	48
Figura 4.8: Módulo de cámara RPI. [44] .....	49
Figura 4.9: Esquema general del sistema .....	54
Figura 4.10: Diagrama de flujo del funcionamiento del sistema .....	55
Figura 4.11: Cargando la imagen de Ubuntu Mate al micro SD.....	56
Figura 4.12: Inicio de la instalación de Ubuntu Mate en RPI 3 .....	57
Figura 4.13: Menú general de Ubuntu Mate .....	57
Figura 4.14: Instalación de eviacam en Ubuntu Mate.....	58
Figura 4.15: EVIACAM detecta el rostro .....	59
Figura 4.16: Instalación de eSpeak .....	61
Figura 4.17: Ejemplo del uso del sintetizador de voz (eSpeak).....	62
Figura 4.18: Comandos usados para instalar PYBLUEZ.....	63
Figura 4.19: Ejemplo para usar el Bluetooth.....	63
Figura 4.20: Ejemplo de prueba de Autopy .....	65
Figura 4.21: Teclado virtual para el paciente.....	66
Figura 4.22: Diagrama de conexión de la interfaz para el paciente. ....	67

Figura 4.23: Conexión de dispositivos a la placa RPI 3.....	69
Figura 4.24: Prototipo final de la interfaz gráfica .....	70
Figura 4.25: Esquema de conexión del sistema domótico .....	71
Figura 4.26: Diagrama eléctrico del sistema domótico.....	72
Figura 4.27: Prototipo final del sistema domótico .....	73
Figura 4.28: Ventana principal del teclado virtual.....	74
Figura 4.29: Sub-ventana de funciones rápidos. ....	75
Figura 4.30: Sub-ventana de control domótico .....	76
Figura 4.31: SMPlayer para escuchar música. ....	77
Figura 4.32: Respuesta mayor obtenida de los encuestados .....	79
Figura 4.33: Resultados obtenidos de personas que usaron el sistema .....	81

## ÍNDICE DE TABLAS

Tabla 2.1: Etapas de ELA. [14].....	10
Tabla 2.2: Clasificación del Bluetooth de acuerdo al alcance. [41].....	23
Tabla 4.1: Velocidad y número de parpadeos .....	29
Tabla 4.2: Análisis técnico y selección del sensor de proximidad infrarrojo .....	30
Tabla 4.3: Comparativa entre mini-ordenadores (SBC).....	32
Tabla 4.4: Características generales de la Raspberry Pi 3 modelo B .....	34
Tabla 4.5: Comparativa entre diferentes placas electrónicas .....	35
Tabla 4.6: Comparativa de las pantallas para la RPI 3.....	37
Tabla 4.7: Comparativa de comunicaciones inalámbricas .....	40
Tabla 4.8: Calibración del sensor .....	43
Tabla 4.9: Potencia total que consume el circuito de los lentes .....	45
Tabla 4.10: Sistemas operativos más comunes. ....	50
Tabla 4.11: Sistemas operativos para la tarjeta Raspberry Pi. ....	51
Tabla 4.12: Librerías y programas empleadas en Python .....	59
Tabla 4.13: Potencia total que consume el circuito de la interfaz gráfica.....	68
Tabla 4.14: Pruebas realizadas en el día y la noche. ....	80
Tabla 4.15: Costo de Hardware y software .....	81
Tabla 4.16: Valor total para implementar el prototipo.....	83
Tabla 4.17: Comparativa del dispositivo comercial y el sistema desarrollado .....	84

## RESUMEN

El presente proyecto de investigación detalla el desarrollo e implementación del prototipo, sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con Esclerosis Lateral Amiotrófica (ELA), esta extraña enfermedad que se presenta generalmente en las personas adultas de 40 a 70 años, destruye las neuronas motoras del cerebro y la médula espinal de forma progresiva, las neuronas motoras del cerebro dejan de enviar impulsos eléctricos a los músculos del cuerpo, provocando la inmovilidad de sus brazos, piernas e incluso la lengua, ocasionando que el paciente no se pueda comunicar con las personas de su alrededor e impidiendo realizar actividades productivas.

El sistema cuenta con un sensor de proximidad ubicado en las lentes del paciente, la cual detecta el movimiento de la comisura lateral del ojo, y envía una señal hacia una interfaz gráfica que está desarrollada en la tarjeta Raspberry PI (RPI).

En la interfaz gráfica se tiene un teclado virtual y una simulación del mouse que se desplaza por cada letra con un tiempo determinado, si el cursor del mouse se encuentra en la letra deseada se puede seleccionar y escribir mediante el movimiento mencionado del ojo, además, al sistema está integrado un altavoz que permite escuchar el mensaje escrito previamente por el paciente. También, se ha generado un pequeño sistema domótico para el control de la luz, puerta, ventana, ventilador y alarma del paciente, las cuales son manipuladas mediante la comunicación Bluetooth.

El sistema adicionalmente cuenta con el módulo de la cámara RPI conectado al puerto CSI, para controlar el cursor del mouse a partir del movimiento de la cabeza, esta opción se integró al sistema debido a que algunos pacientes están en etapas intermedias y cuentan con movilidad de su cabeza, de esta manera el proceso de escritura y el control de dispositivos externos se realizará con mayor rapidez.

**Palabras clave:** ELA, Raspberry Pi, Python, Tkinter, Domótico.



## **ABSTRACT**

This research project details the development and implementation of the prototype, electronic system to facilitate communication and daily activities of patients with Amyotrophic Lateral Sclerosis (ALS), this strange disease that usually occurs in adults aged 40 to 70 years destroys the motor neurons of the brain and the spinal cord progressively, the motor neurons of the brain stop sending electrical impulses to the muscles of the body causing the immobility of their arms, legs and even the tongue, causing the patient can't communicate with the people around you and even preventing productive activities.

The system has a proximity sensor located in the patient's glasses, which detects the movement of the lateral commissure of the eye, and sends a signal to a graphical interface that is developed on the Raspberry PI (RPI) card.

In the graphical interface you have a virtual keyboard and a mouse simulation that moves for each letter with a certain time, if the mouse cursor is in the desired letter can be selected and written by the movement of the lateral corner of the eye, in addition to the system is integrated a loudspeaker that allows to listen to the message previously written by the patient. Also, a small home automation system has been generated to control the light, door, window, ventilator and patient alarm, which are manipulated through Bluetooth communication.

The system also has the module of the RPI camera connected to the CSI port, to control the mouse cursor from the movement of the head, this option was integrated into the system because some patients are in intermediate stages and have mobility of your head, in this way the writing process and the control of external devices will be done more quickly.

**Keywords:** ALS, Raspberry Pi, Python, Tkinter, Domotic

## **GLOSARIO DE TÉRMINOS Y ACRÓNIMOS**

**ASCII** (American Standard Code for Information Interchange) Código Estándar Estadounidense para el Intercambio de Información.

**AVR** (Advanced Virtual Rise) Microcontrolador fabricado por ATMEL con juego de instrucciones reducido (RISC).

**CDC** (Connected Device Configuration) Configuración de Dispositivos Conectados

**CLDC** (Connected, Limited Device Configuration) Configuración limitada de dispositivos conectados.

**CSI** (Camera Serial Interface) Interfaz Serie para Cámaras.

**GPIO** (General Purpose Input/Output) Entrada/Salida de Propósito General.

**GNU** (General Public License) Licencia pública general.

**GUI** (Graphical User Interface) Interfaz Gráfica de Usuario.

**HDMI** (High-Definition Multimedia Interface) Interfaz multimedia de alta definición

**IDE** (Internal Development Environment) Entorno de desarrollo nativo o plataforma de programación.

**ISO** (Organization for Standardization) Organización Internacional de Normalización.

**IVR** (Interactive Voice Response) Respuesta de Voz Interactiva

**RHEL** (Red Hat Enterprise Linux también) Es una distribución comercial de Linux desarrollada por Red Hat.

**PDA** (Personal Digital Assistant) Asistente Digital Personal

**PWM** (Pulse Width Modulation) Modulación por ancho de pulso.

**RAM** (Random Access Memory) Memoria de Acceso Aleatorio.

**SD** (Secure Digital) Dispositivo en formato de tarjeta de memoria para dispositivos portátiles.

**TTL** (Transistor – Transistor Logic) Lógica transistor – transistor. Más conocida como lógica de 5V.

**USB** (Universal Serial Bus) Conector serial universal

**USSD** (Unstructured Supplementary Service Data) Servicio Suplementario de Datos no Estructurados, es un protocolo para el envío de datos a través de móviles GSM).

**VDI** (Virtual Desktop Infrastructure) Es la práctica de alojar un sistema operativo de escritorio dentro de una máquina virtual (VM).

## INTRODUCCIÓN

En la actualidad la enfermedad de ELA (Esclerosis Lateral Amiotrófica), se ha incrementado considerablemente a nivel mundial, las personas que padecen de este tipo de malestar en las etapas finales soportan parálisis completa de su cuerpo a excepción de los ojos, los parpados y en ocasiones la cabeza, es por ello que varias empresas desarrollan diversos dispositivos electrónicos para ayudar a que estos pacientes se comuniquen y puedan realizar alguna actividad sin depender totalmente de las personas que lo asisten.

En el presente proyecto se implementa el prototipo, sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con ELA.

El proyecto de investigación consta de cinco capítulos, los cuales se describen brevemente a continuación:

En el capítulo uno, se expone el análisis completo de la enfermedad de ELA conociendo las etapas iniciales y etapas finales de la parálisis del cuerpo, estos aspectos ayudan al desarrollo de la investigación con planteamiento del problema, la justificación del porque realizar el sistema electrónico y trazando objetivos que encaminen el desarrollo del proyecto.

En el capítulo dos, se detallan las distintas maneras de comunicación aumentativa y alternativa, así como los diferentes dispositivos de ayuda para los pacientes con esta enfermedad y finalmente se analiza la propuesta con la cual se dará solución al problema planteado.

En el capítulo tres, se define la metodología, el tipo de investigación, recolección de la información y las actividades que se desarrollaron para la ejecución del proyecto de investigación.

En el capítulo cuatro, se describe de manera detallada el diseño e implementación del prototipo sistema electrónico para facilitar la comunicación y actividades cotidianas de los pacientes con ELA.

En el capítulo cinco, señala las conclusiones y recomendaciones obtenidas al finalizar el proyecto de investigación.

## **CAPÍTULO 1**

### **EL PROBLEMA**

#### **1.1 Tema**

“Sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con Esclerosis Lateral Amiotrófica (ELA).”

#### **1.2 Planteamiento del problema**

El gran inconveniente que atraviesa la sociedad actualmente radica en la presencia de diversos tipos de enfermedades, las cuales atacan e incluso acaban con la vida de las personas, como es el caso de ELA (Esclerosis Lateral Amiotrófica, también conocido como la enfermedad de Lou Gehrig y del científico Stephen Hawking), es una enfermedad neurodegenerativa progresiva que afecta a las neuronas motoras del cerebro, las cuales son las encargadas de controlar el movimiento de los músculos. ELA lesiona las neuronas motoras e impide que el cerebro envíe impulsos eléctricos a la médula espinal, paralizando la mayor parte de los músculos del cuerpo y provocando que la persona afectada no pueda comunicar e incluso realizar actividades cotidianas. [1]

En Inglaterra y Australia de acuerdo al artículo presentado en el año 2014 por la Fundación Española para el Fomento de la Investigación de la Esclerosis Lateral Amiotrófica (FUNDELA), a esta enfermedad también se le conoce como “enfermedad de la moto neurona” (EMN) o enfermedad de la neurona motora (ENM), pero comúnmente es conocido como ELA, esta enfermedad afecta en mayoría de los casos a las personas adultas de 40 a 70 años, aunque también se puede presentar fácilmente en los jóvenes. En Europa

específicamente en España se diagnostican casi unos 900 nuevos casos de ELA de 2 a 3 por día y que el número total de casos ronda las 4000 personas, aunque estas cifras pueden variar, la incidencia de esta enfermedad en la población española es de 1 por cada 50.000 habitantes. Sin embargo, en la mayoría de los países de Europa no se ha encontrado un sistema para ayudar a los enfermos con ELA y los pocos dispositivos que existen son realizados por grandes empresas, la desventaja de estos productos son los costos elevados lo cual impide que muchas personas de bajos recursos puedan adquirir fácilmente estos dispositivos de ayuda para pacientes con ELA. [2]

La investigación realizada en el Ecuador, ha comprobado que en el año 2014 la enfermedad de ELA era casi desconocida ya que se presentaron únicamente 5 casos, además se considera que existe un subdiagnóstico de esta patología debido a que en el país no existe la preparación suficiente en los médicos para detectar este tipo de enfermedad. Actualmente según la Fundación Ecuatoriana de Esclerosis Múltiple (FUMDEM), en el año 2016 se ha presentado 50 casos de ELA, lo cual es preocupante porque se puede verificar que la enfermedad se va incrementando considerablemente y se requiere realizar un registro nacional de ELA de manera urgente. Adicionalmente, las estadísticas señalan que la patología en el Ecuador se presenta en personas entre los 25 a 40 años, contrario a las cifras mundiales que indican que los síntomas generalmente se manifiestan hasta después de los 40 y 70 años. Las personas que padecen de este tipo de enfermedad en las etapas finales soportan parálisis completa de su cuerpo a excepción de los ojos y los párpados, sin embargo, en el Ecuador no existe un dispositivo que facilite la comunicación de los pacientes con ELA. Por lo tanto, se ha visto necesario realizar un dispositivo electrónico para ayudar a que las personas que padecen esta enfermedad se comuniquen, debido a que únicamente pueden mover sus párpados. [3]

### **1.3 Delimitación**

#### **Delimitación de contenidos:**

Área académica: Física y Electrónica.

Líneas de investigación: Sistemas de Control

Sublínea de Investigación: Sistemas Embebidos

## **Delimitación Espacial**

La presente investigación se desarrolló en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial enfocado el estudio a las personas que padecen la enfermedad de ELA en el Ecuador.

## **Delimitación Temporal**

El proyecto de investigación se desarrolló durante el periodo Marzo – Septiembre del 2017, de acuerdo a lo establecido en el Reglamento de Graduación para Obtener el Título Terminal de Tercer nivel en la Universidad Técnica de Ambato.

### **1.4 Justificación**

En el Ecuador la enfermedad de ELA se ha incrementado en los 2 últimos años, por esta razón es importante la realización del proyecto de investigación, debido a que presenta el diseño y desarrollo adecuado del sistema electrónico para brindar una alternativa de comunicación a un grupo específico de personas que padecen esta enfermedad, la cual impide adaptar a un sistema de comunicación común, del mismo modo se pretende ayudar a estas personas para que su autoestima se eleve y pueden dependerse de sí mismos realizando actividades productivas.

Con el desarrollo de este proyecto los beneficiarios directos serán las personas que padecen la enfermedad de ELA, debido a que pierden movimientos voluntarios de la mayoría de su cuerpo, incluso de la lengua y se les dificulta la comunicación con las personas de su alrededor, además los beneficiarios de forma indirecta son los familiares cercanos de los pacientes ya que es difícil estar pendiente la mayoría del tiempo y la frustración aumenta, pero mediante el sistema planteado se pretende facilitar la vida cotidiana que llevan estas familias.

En consecuencia el presente proyecto será realizado en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial (FISEI), puesto a que se requiere de los equipos y laboratorios necesarios para el desarrollo de la misma, la tecnología que se pretende emplear es de Open Source (Software y Hardware libre), que es ampliamente administrable y tiene la ventaja en la reducción de costos para el desarrollo del

sistema, de esta manera se brinda facilidades para que las personas de bajos recursos económicos puedan adquirir el dispositivo.

## **1.5 Objetivos**

### **1.5.1 Objetivo General**

Implementar el prototipo del sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con Esclerosis Lateral Amiotrófica (ELA).

### **1.5.2 Objetivos Específicos**

- Analizar el movimiento de los parpados de los ojos de las personas que padecen la enfermedad de ELA.
- Determinar el funcionamiento de dispositivos electrónicos para ayudar a los pacientes con ELA.
- Diseñar el sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con ELA.



## CAPÍTULO 2

### MARCO TEÓRICO

#### 2.1 Antecedentes Investigativos

Al finalizar la investigación basados en artículos científicos, investigación de proyectos realizados a nivel comercial y académica se pueden encontrar los siguientes antecedentes:

La empresa INTEL, en el año 2015 presento modificaciones del “*Sistema ACAT (Assistive Context-Aware Toolkit) Conjunto de Herramientas para el Contexto Asistido*”, el cual es usado por el profesor Hawking y consiste en capturar los movimientos del músculo de la mejilla, estos movimientos se traducen en palabras, que son dictadas en tiempo real por el programa de síntesis de voz. ACAT no sólo es un sistema de predicción, también incluye un teclado virtual para editar texto, tiene la capacidad de abrir documentos y navegar en internet. Además, es compatible con una amplia gama de dispositivos como cámaras web, sensores y otros periféricos. Parte importante de este sistema es que ACAT ahora ya es de código abierto y es compatible con Windows a partir de XP, la programación está realizada en C# y su código se puede encontrar en Intel Open Source, donde cualquier desarrollador tendrá al alcance todo lo necesario para la creación de nuevas herramientas. [4]

La empresa sueca Tobii Dynavox, en el año 2015 desarrollo “*Dispositivos que lee el movimiento de las pupilas*”, estos dispositivos cuentan con una pantalla táctil y un lector ocular que detecta el movimiento de las pupilas. El usuario puede, entonces, interactuar con la pantalla, jugar, navegar por Internet o usar un teclado para escribir según el potencial de cada persona afectada por ELA. Tobii Dynavox además ofrece un método

directo de acceso es decir no hace falta barrido como es el caso del sistema de Hawking, se activa sólo con mirar la tecla en pantalla. Los sistemas que presenta Tobii Dynavox se basan en instalar en una Tablet un sistema operativo desarrollado por la propia empresa y mediante la Webcam y sensores infrarrojos se puede sincronizar y ajustar el movimiento de las pupilas del ojo, de esta manera se tiene el control total de la Tablet a modo de un mouse, la otra forma de estos dispositivos es adaptar una unidad en la máquina, instalar sus Drivers y sincronizar al ojo del paciente para que pueda controlar la PC. [5]

La empresa de Israel en el año 2015 ha creado el “*Dispositivo que puede cambiar la historia de los pacientes con ELA (EyeControl)*”, el cual se basa en un dispositivo laser conectado a unas gafas. EyeControl utiliza una cámara de infrarrojos conectada a un simple par de lentes para identificar la dirección del ojo, hacia donde está mirando, si parpadea, y controlar sus movimientos. La cámara de infrarrojos se comunica con un ordenador llamado “ODROID” a través de USB. A continuación, el “ODROID” traduce la dirección del ojo del usuario y sus movimientos en comandos.

Los comandos pueden ser comunicados en tres salidas, auriculares, altavoces, y un teléfono inteligente a través de Bluetooth, para transmitir el comando, la aplicación móvil de EyeControl utiliza un algoritmo único para calibrar el dispositivo según las necesidades del paciente. [6]

Marlon Fernando Vaca Muñoz, en la Universidad de las Fuerzas Armadas ESPE, sede Sangolqui, en el año 2016 desarrollo el “*Diseño, construcción e implementación de un sistema móvil y de elevación automático de una silla de ruedas para personas con esclerosis lateral amiotrófica (ELA)*”, el cual es un sistema móvil que permite mediante mecanismos de bipedestación, evitar trastornos sobre la columna vertebral al momento de trasladar al paciente de la silla a otro lugar. El mecanismo de reclinación logra permutar posiciones, entre sentado y acostado que colabora con el bienestar de la salud. Los mecanismos de elevación y reclinación usan motores eléctricos lineales como mejor solución para solventar el problema y por su facilidad de control, el mecanismo de movilidad automática permite mejorar el nivel de independencia que posee el paciente. El conjunto de estas tres características ayuda a confrontar los problemas del uso de la silla de ruedas, El sistema de control posee lógica difusa para solventar dificultades de varias variables de entradas y variables de salida. El sistema mecánico

fue dimensionado correctamente para funcionar con una persona de peso promedio, para facilitar al usuario el manejo de la silla de ruedas, se implementó una interface de usuario HMI, con una pantalla táctil para el control general y con un mando analógico (joystick) para los controles manuales del sistema. [7]

María Gabriela Vintimilla Sarmiento, en la Universidad de las Fuerzas Armadas ESPE, sede Sangolqui, año 2014 presento el artículo científico con el tema “*Desarrollo e Implementación de una Aplicación que Traduzca el Abecedario y los Números del uno al diez del Lenguaje de Señas a Texto para Ayuda de Discapacitados Auditivos Mediante Dispositivos Móviles Android.*”, este artículo se basa en el diseño e implementación de una aplicación para dispositivos móviles con sistema operativo Android, que permita la interrelación de las personas con discapacidad auditiva. La aplicación es capaz de aprender y reconocer una letra o número del lenguaje de señas que no tengan movimiento mediante la aplicación de redes neuronales artificiales, la aplicación indicara si la imagen captada pertenece a la letra que se desea reconocer, en caso de que la imagen no pertenezca a la letra correspondiente, la aplicación muestra un mensaje de error. [8]

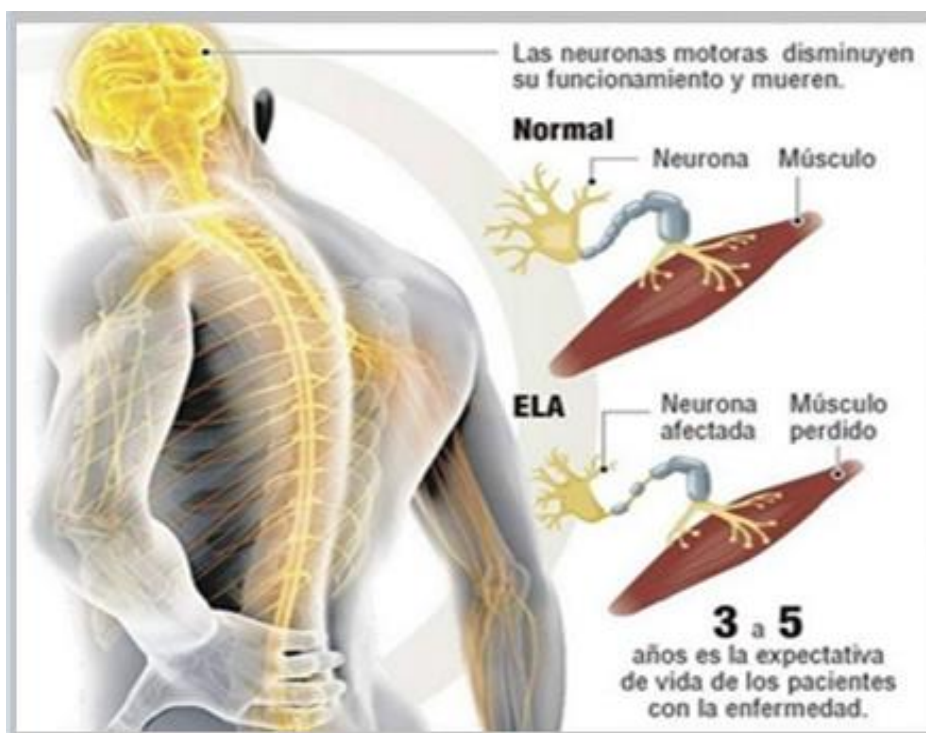
Dina Patricia Jarrin Sambrano, Escuela Politécnica Nacional desarrollo el “*Mouse para Personas con Discapacidad Motriz*”, El propósito de este proyecto es permitir que personas con discapacidad motriz, tengan la posibilidad de utilizar el mouse del computador. Se utiliza una cámara web de bajo costo, se manipula un objeto reflector para evitar problemas de iluminación y facilita su detección mediante el procesamiento digital de imágenes, de modo que su seguimiento emule el puntero del mouse. Además, se estudia alternativas para la activación tanto del botón derecho como del izquierdo del mouse. [9]

La información recopilada de los artículos científicos, repositorios de Universidades, y de páginas destacadas del Internet respecto a la Esclerosis Lateral Amiotrófica (ELA) y dispositivos de apoyo para esta enfermedad, fueron de vital importancia para el desarrollo del proyecto de una manera óptima, debido a que es un sistema electrónico interesante e importante.

## 2.2 Fundamentación teórica

### 2.2.1 Esclerosis Lateral Amiotrófica (ELA)

La Esclerosis Lateral Amiotrófica (ELA), es una enfermedad neurodegenerativa progresiva que destruye a las neuronas motoras del cerebro y la médula espinal, las neuronas dejan de enviar impulsos eléctricos a los músculos causando la inmovilidad de las piernas, los brazos, en incluso la lengua perdiendo casi en su totalidad el movimiento del cuerpo. Sin embargo, ELA no afecta a los sentidos del paciente como el ver, oír, oler y tocar, del mismo modo esta enfermedad no presenta ninguna posibilidad de que sea contagiosa. En la figura 2.1 se muestra como la enfermedad de ELA afecta los músculos. [10]



*Figura 2.1: Esclerosis Lateral Amiotrófica (ELA). [11]*

### Causas de ELA

ELA es una enfermedad muy extraña que incluso los mismos especialistas no tienen una explicación clara de cuáles son las causas, sin embargo, en la mayoría de los casos aproximadamente en un 90% la ELA se presenta de forma esporádica y sin origen conocido, pero existe también un tipo de ELA familiar conocido como de carácter hereditario. [12]

## **Diagnósticos de ELA**

La enfermedad se diagnostica mayoritariamente entre los 40 y 70 años de edad, aunque puede afectar a un rango mucho más amplio de edades, y es más frecuente en varones. De acuerdo a las investigaciones realizadas por la plataforma de afectados por ELA de España, cada año se diagnostican 2 casos de ELA por cada 100.000 habitantes. La esperanza de vida después del diagnóstico es de 5 años para el 80% de los pacientes y para el porcentaje restante no se han presentado diagnósticos exactos debido a que la evolución de la enfermedad puede variar de acuerdo a las características de cada paciente, en algunos casos la Esclerosis Lateral Amiotrófica se desarrolla más rápido mientras que en otros casos el desarrollo es lento, por esta razón los pacientes con ELA pueden vivir más tiempo del diagnóstico presentado. [12]

## **Síntomas iniciales de ELA**

La ELA comienza detectándose como una pérdida de la fuerza muscular en brazos y piernas, impidiendo hacer acciones cotidianas. También puede aparecer por primera vez en los músculos de la boca, impidiendo la correcta respiración o capacidad de deglutir o tragar. La enfermedad se va extendiendo por la corteza cerebral y la médula espinal hasta que, en las etapas finales, se producen la pérdida de movilidad absoluta y el fallo respiratorio, que suele ser la causa de fallecimiento más habitual en esta patología. [13]

### **Los síntomas más comunes son:**

- Problemas para la deglución y el habla, especialmente, habla lento o dificultades en la articulación de las palabras.
- Sensación de cansancio y pesadez excesiva en los músculos controlados por la médula espinal.
- Dificultades para levantar objetos, subir escaleras y caminar.
- Rigidez muscular o calambres.
- Contracciones musculares (fasciculaciones).
- Cambios en la voz (ronquera).
- Parálisis muscular progresiva.
- Pérdida de peso.

## Diferentes etapas de ELA

La enfermedad de ELA afecta a los músculos del cuerpo, en consecuencia, la movilidad de los músculos varía de acuerdo a la condición física de cada paciente, sin embargo, las etapas que generalmente se presentan en esta enfermedad son cuatro, como se puede apreciar en la tabla 2.1. [14]

*Tabla 2.1: Etapas de ELA. [14]*

<b>ETAPAS</b>	<b>SÍNTOMAS</b>
Inicial o temprana	Los pacientes comienzan a experimentar la leve debilidad muscular. Los dedos todavía están en funcionamiento, pero el nivel de destreza está cayendo.
Media	Afecta la debilidad muscular progresiva brazos y piernas. Es difícil mover brazos, manos y el cuerpo.
Tardía	Presenta mayor debilidad muscular y atrofia extendida a otras partes del cuerpo. Los brazos y las piernas ya no se pueden mover. Los pacientes son capaces de comunicarse por expresiones faciales emocionales.
Final	Pese a que los músculos más voluntarios están sin movilidad, el movimiento voluntario de ojos sigue siendo funcional. La velocidad de movimiento de los ojos puede ser lento.

## Tratamientos de la enfermedad de ELA

La ELA es una enfermedad que no tiene tratamiento curativo. Sin embargo, la terapia que se les oferta a los pacientes es para disminuir los síntomas que irán apareciendo con el paso del tiempo.

La terapia que se les ofrece es solventar los problemas derivados de la atrofia muscular y hacer frente a la posible depresión que pueda surgir del estado del paciente. Para hacer frente al dolor se dispone de fármacos que se suelen utilizar de manera conjunta, ya que

el dolor que presentan suele tener diferentes orígenes. El dolor suele ser consecuencia de la reducción de la movilidad, los calambres y la espasticidad (aumento del tono muscular). El medicamento que se usa son el antiinflamatorio no esteroideo, opiáceos, cannabinoides y fármacos para el dolor de origen neuropático.

Ejerce un papel importante dentro del tratamiento para el dolor la fisioterapia, que además de terapéutico sirve como prevención si se comienza con ella en los estados más iniciales de la enfermedad. [15]

### **2.2.2 Sistemas de Comunicación Aumentativa y Alternativa (SAAC)**

Son formas de expresión distintas al lenguaje hablado, que tienen como objetivo aumentar (aumentativos) y/o compensar (alternativos) las dificultades de comunicación y lenguaje de muchas personas con discapacidad. Entre las causas que pueden hacer necesario el uso de un SAAC encontramos la parálisis cerebral (PC), la discapacidad intelectual, los trastornos del espectro autista (TEA), las enfermedades neurológicas tales como la Esclerosis Lateral Amiotrófica (ELA), la esclerosis múltiple (EM) o el párkinson, las distrofias musculares, los traumatismos cráneo-encefálicos, las afasias o las pluridiscapacidades de tipologías diversas, entre muchas otras. [16]

### **Estrategias para la Comunicación Aumentativa y Alternativa**

Gracias a las diferentes estrategias y productos de apoyo para el acceso, por muy restringida que se encuentre la movilidad de una persona, casi siempre es posible encontrar una solución para que pueda acceder a la comunicación y a otras actividades, tales como la movilidad asistida, el control del entorno o el acceso al ordenador para la escritura, el dibujo, el juego o la comunicación a través de la red.

Las estrategias del SAAC son. [16]

#### **a) La selección directa**

Consiste en señalar o pulsar las teclas directamente, con el dedo, con la mirada o con otras partes del cuerpo, para indicar los pictogramas, palabras o letras que se quieren comunicar. Los punteros de distinto tipo son ejemplos de productos de apoyo que puede facilitar la selección o acceso directo.

### **b) La selección con ratón**

Son diseñados específicamente para productos electrónicos, consiste en acceder con un ratón a teclados o cuadrículas con símbolos para la comunicación en pantalla. Se puede usar una gran variedad de ratones adaptados, en forma de joystick, trackball, así como el ratón facial (controlado con movimientos de la cabeza), el ratón controlado con la mirada o el multimouse que consistente en cinco teclas o conmutadores.

### **c) La exploración o barrido**

Se usa en tableros o libros, consiste en que el interlocutor vaya señalando, uno en uno o por grupos, filas y columnas, los símbolos o letras a comunicar, hasta que el hablante asistido indique con un gesto que se ha dado con el que quería comunicar.

### **d) La exploración o barrido independiente**

Solamente para productos electrónicos, en este caso es el comunicador u ordenador el que presenta las diferentes opciones a comunicar hasta que el hablante asistido selecciona la que le conviene pulsando un conmutador. Existen muchos tipos de conmutadores que se pueden activar con diferentes partes del cuerpo.

### **e) La selección codificada**

En este caso cada símbolo o letra tiene un código, por ejemplo, un número de dos o tres cifras o un color y un número, de manera que el hablante asistido indica de forma directa o por barrido este código para transmitir el símbolo o letra. De esta forma con pocas teclas o casillas puede acceder a un gran número de símbolos.

## **2.2.3 Dispositivos Electrónicos de Comunicación para enfermos con ELA**

En los últimos años el mundo y el impacto tecnológico han cambiado drásticamente facilitando de cierta manera a muchas personas con distintos tipos de parálisis y discapacitados se puedan integrar a la sociedad actual. La medicina hasta el momento no ha podido con la ELA, entonces la tecnología está en la obligación de intervenir, es así que empresas potentes desarrollan diversos tipos de dispositivos electrónicos con la finalidad de ayudar a los pacientes con ELA, entre los más destacados están el ACAT



el cual fue desarrollado específicamente para el profesor Hawking, EyeControl es un dispositivo que escribe mediante el movimiento de los ojos, Tobbi Dynavox y muchas dispositivos facilitan la vida diaria que llevan dichos pacientes. [17], [18]

### **Características**

- Permiten manejar una PC con el movimiento de las pupilas a partir de las luces infrarrojas conectadas al ordenador.
- Desarrollan aplicaciones Android para escribir con teclados intuitivos y transformar a voz los mensajes que se escriben.
- Facilitan el control de dispositivos externos (Domótica) con el movimiento de los ojos.
- Desarrollan dispositivos específicos de entretenimiento como aplicaciones de juegos.

### **Sistema ACAT**

El sistema ACAT (Assistive Context-Aware Toolkit) desarrollado por la empresa Intel para el científico Stephen Hawking quien padece de la enfermedad de ELA, es uno de los más grandes avances para hacer que los ordenadores sean accesibles a las personas con discapacidad.

El sistema ACAT, consiste en capturar los movimientos del músculo de la mejilla, estos movimientos se traducen en palabras, que son dictadas en tiempo real por el programa de síntesis de voz. Parte importante de este sistema, es que ha sido creado especialmente para Hawking, gran parte está basado en el sistema de predicción SwiftKey. Además, ACAT no sólo es un sistema de predicción, también incluye un teclado virtual para editar texto, tiene la capacidad de abrir documentos y navegar en internet. Además, es compatible con una amplia gama de dispositivos como cámaras web, sensores y otros periféricos.

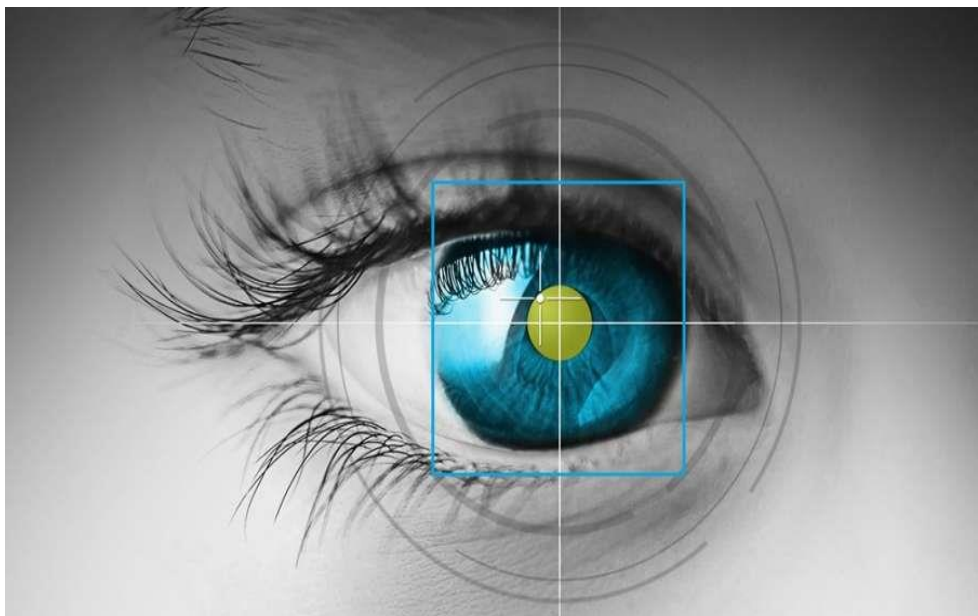
El sistema ha sido liberado para que cualquier persona lo pueda descargar y usar sin coste, ACAT es ahora una plataforma Open Source. En la figura 2.2 se muestra el sistema ACAT utilizado por el científico Stephen Hawking. [19]



*Figura 2.2: Sistema ACAT de Stephen Hawking. [20]*

### **Sistema Eye-Traking**

Es un dispositivo o computadora equipada con un rastreador ocular. Esto hace posible que los usuarios interactúen con el mouse de la computadora usando sus ojos de manera sencilla. En la figura 2.3 se muestra el seguimiento ocular con Eye-Traking. [21]



*Figura 2.3: Seguimiento ocular Eye-Traking. [22]*

## Tobii Dynavox

Es una empresa que desarrolla dispositivos de comunicación estructuradas para atender o cubrir las necesidades específicas de las personas con: autismo, parálisis cerebral, síndrome de Down, discapacidades del desarrollo, así como otros problemas de comunicación. En la figura 2.4 se puede apreciar el dispositivo creado por la empresa Tobii Dynavox. [23]



Figura 2.4: Dispositivo desarrollado por Tobii Dynavox. [24]

### 2.2.4 Sistema Electrónico

Un sistema electrónico es un conjunto organizado que transforma una señal de entrada en una señal de salida, de acuerdo al principio de la transformación. Dicha señal puede ser de forma analógica o digital.

En todo sistema electrónico se tiene dispositivos de entrada, dispositivos de proceso y dispositivos de salida como se ilustra en la figura 2.5. [25], [26]

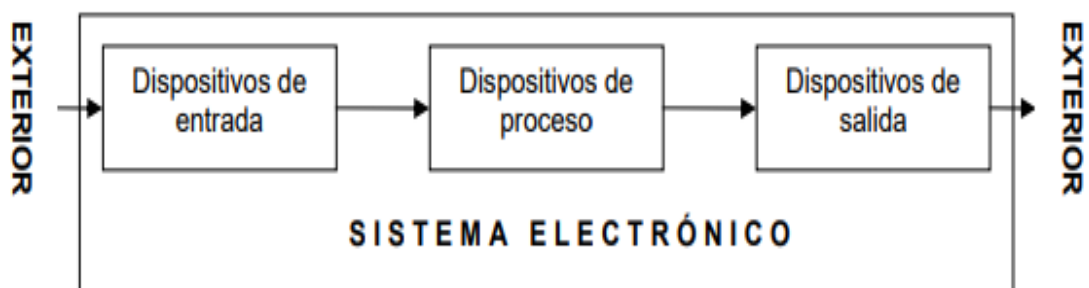


Figura 2.5: Sistema Electrónico [26]

## **a) Dispositivos de Entrada**

Los dispositivos de entrada son sensores o transductores electrónicos y mecánicos que toman las señales externas en forma de temperatura, humedad, presión, movimiento y las convierten en señales de corriente o voltaje. [26]

### **Sensor**

Es un dispositivo que, a partir de la energía del medio donde se mide, da una señal de salida transductible que es función de la variable de medida. La ampliación de los sentidos para adquirir un conocimiento de cantidades físicas que, por su naturaleza o tamaño, no pueden ser percibidas directamente por los sentidos. [27]

### **Sensor de proximidad**

Un sensor de proximidad es un sensor electrónico capaz de detectar la presencia de objetos cercanos sin contacto físico. Emite un campo electromagnético o un haz de radiación electromagnética (por ejemplo, infrarrojo) y luego busca cambios en la señal o campo de retorno. El objeto que se está detectando es el objetivo del sensor de proximidad. [28]

### **Tipos de sensores de proximidad**

Existen diversos tipos de sensores de proximidad según el principio físico que utilizan, los más comunes son: [29]

- Sensor capacitivo
- Sensor inductivo
- Sensor fin de carrera
- Sensor infrarrojo
- Sensor ultrasónico
- Sensor magnético

### **Sensor infrarrojo**

El sensor de infrarrojos es un sensor de medición de distancia, que se basa en un sistema de emisión/recepción de radiación lumínica en el espectro de los infrarrojos (menor que

las ondas de radio y mayor que la luz). Una de las técnicas más habituales para la medición de la distancia es mediante la triangulación del haz de luz colimada, si bien también se puede considerar la distancia de un objeto a partir de la cantidad de energía recibida tras rebotar la luz sobre un objeto.

En robótica y aplicaciones prototipos se suelen utilizar sensores muy económicos de corto alcance, en un rango máximo de unos 50/80 cm. y el tipo de detección que realizan es direccional, es decir, sólo son capaces de detectar objetos que están enfrente del sensor. [30]

## **b) Dispositivos de Proceso**

Los dispositivos de proceso son los dispositivos electrónicos encargados de manipular, interpretar y transformar las señales de voltaje y corriente provenientes de los transductores. [31]

Entre los dispositivos de proceso que más se emplean en la electrónica moderna se puede mencionar los siguientes:

- Raspberry Pi
- BeagleBone
- JaguarBoard
- Odroid-C2

### **Raspberry Pi**

La Raspberry Pi es un ordenador de placa reducida, ordenador de placa única o computador de placa simple (SBC) de bajo costo, el tamaño es similar a una tarjeta de crédito, fue desarrollado en reino Unido por la fundación Raspberry PI (Universidad de Cambridge) en el 2011, con el objetivo de estimular la enseñanza de la informática en las escuelas. [32]

### **BeagleBone**

El BeagleBone es una mini computadora, del tamaño de una tarjeta de crédito. Muy similar a una Raspberry Pi. Se puede conectar un mouse, un teclado y una pantalla, y

tendrá una computadora completamente funcional. Tiene diversas aplicaciones tanto para videos juegos como para aplicaciones de la robótica o sistemas de automatización para la casa. [33]

### **Jaguarboard**

Esta placa se caracteriza por ser una de las primeras y únicas opciones que montan un procesador X86 en lugar de un procesador ARM, lo que permite funcionar prácticamente igual que un ordenador convencional y ejecutar los mismos programas que este. Permite instalar prácticamente cualquier sistema operativo que instalaríamos en un ordenador convencional, aunque, oficialmente, se da soporte a Windows, Ubuntu, CentOS y Fedora. [34]

### **c) Dispositivos de Salida**

Los dispositivos de salida son los dispositivos electrónicos encargados de convertir las señales de voltaje o corriente en señales físicamente útiles. Los dispositivos de salida pueden ser los actuadores.

### **Actuadores**

Los actuadores son dispositivos mecánicos que brindan la posibilidad de transformar energía para generar el funcionamiento dentro de un sistema automatizado determinado. Estos generan una fuerza a partir de distintos elementos, como puede ser energía eléctrica, líquido o bien en estado gaseoso los cuales son su fuente de energía.

Se puede clasificar a los actuadores en diferentes clases: [31]

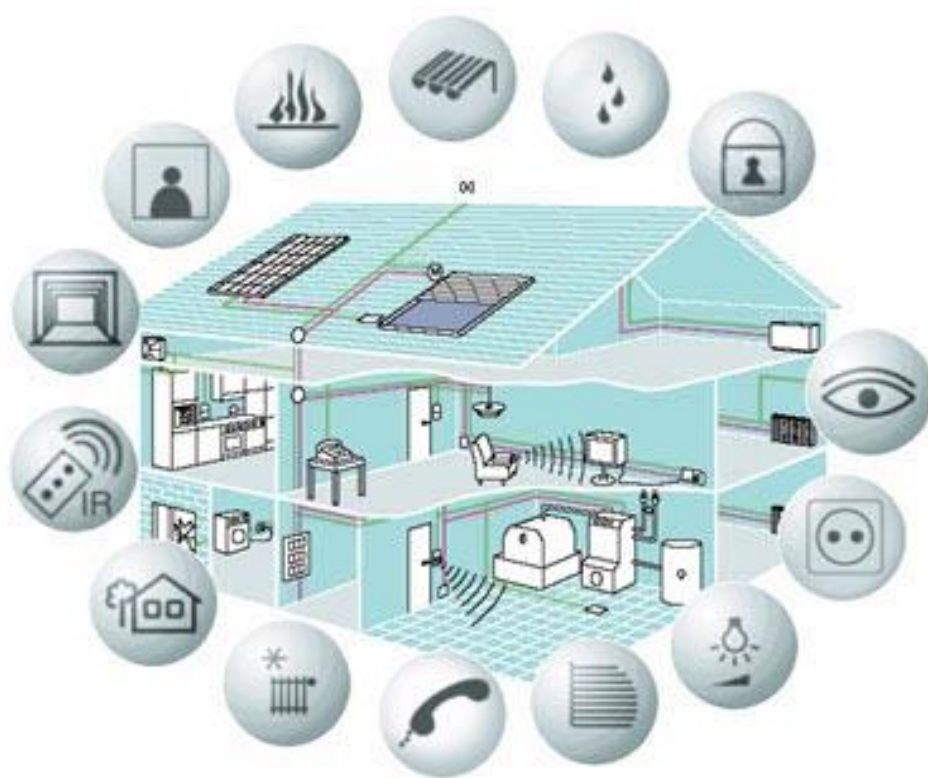
- Hidráulicos
- Electrónicos
- Eléctricos

### **2.2.5 Domótica**

La domótica es el conjunto de sistemas que automatizan las diferentes instalaciones de una vivienda, es un conjunto de tecnologías aplicadas al control y la automatización

inteligente de la vivienda, que permite una gestión eficiente del uso de la energía, además de aportar seguridad, confort, y comunicación entre el usuario y el sistema.

El principal objetivo de estas tecnologías es la mejora de la calidad de vida incrementando la comodidad de las personas; la domótica se puede implementar en distintos ámbitos del hogar. Desde los típicos controladores de la calefacción hasta sistemas de gestión del agua, iluminación, gas o incluso sistemas automáticos de persianas y toldos basados en sensores de incidencia solar y temperatura. Todos ellos para conseguir ahorrar agua, gas y, sobre todo electricidad. En la figura 2.7 se muestra la automatización domótica de una vivienda. [35], [36]



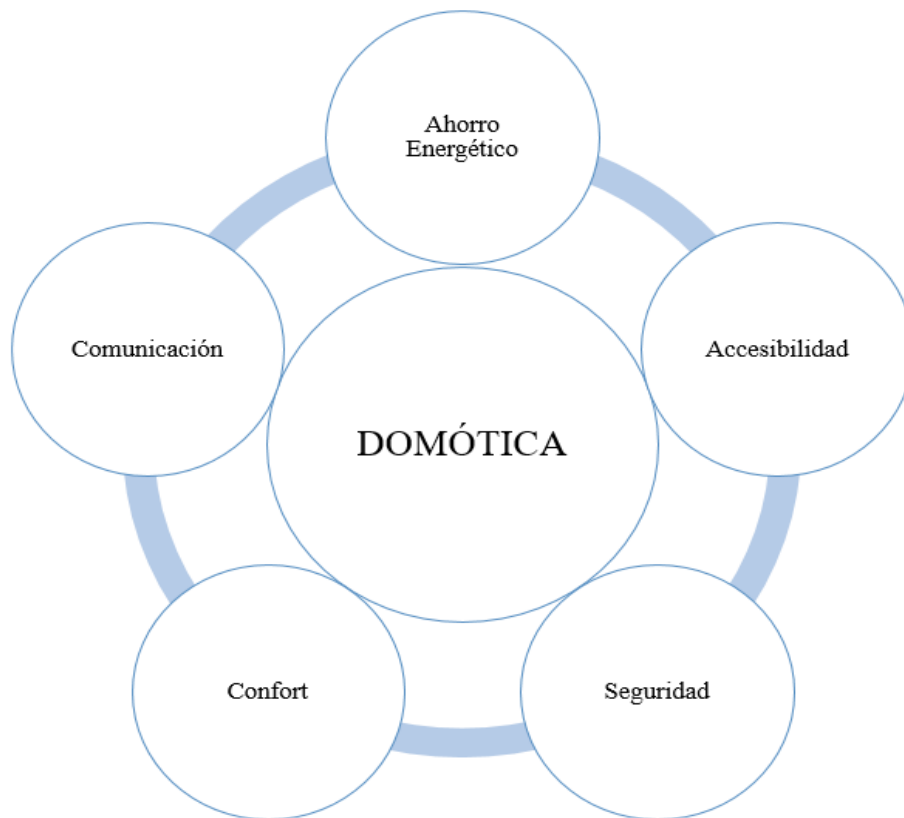
*Figura 2.6: Automatización domótica de una vivienda. [37]*

### **Aportes de la domótica**

La domótica contribuye a mejorar la calidad de vida del usuario, como se muestra en la figura 2.8. [38]

- Facilitando el ahorro energético.
- Fomentando la accesibilidad.
- Aportando la seguridad mediante la vigilancia automática de personas.

- Convierte la vivienda en lugar más confortable.
- Garantiza la comunicación mediante la comunicación y supervisión remota.



*Figura 2.7: Aportes de la domótica*

*Elaborado por: Investigador*

### **2.2.6 Software libre**

Software Libre es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. [39]

#### **Libertades esenciales**

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

- La libertad de ejecutar el programa como se desea, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y cambiarlo para que haga lo que usted quiera (libertad 1). El acceso al código fuente es una condición necesaria para ello.



- La libertad de redistribuir copias con la finalidad de ayudar a su prójimo, al vecino o a cualquiera (libertad 2).
- La libertad de distribuir copias de sus versiones modificadas a terceros (libertad 3). Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

### **2.2.7 Comunicación inalámbrica**

La comunicación inalámbrica es un método de transmisión de cualquier tipo de información de un punto a otro, sin utilizar ninguna conexión como cables, es decir, la información se transfiere desde el transmisor hacia el receptor utilizando ondas electromagnéticas, el medio de propagación puede ser el aire o la atmosfera. [40]

#### **Tipos de sistemas de comunicación inalámbrica**

En la actualidad las personas necesitan teléfonos móviles para muchas cosas como llamar, multimedia, etc. Todos estos servicios deben estar disponibles para el usuario mientras viaja, es decir, mientras el usuario es móvil. Además se requiere de servicios inalámbricos para el hogar ya sea para acceder al internet o controlar algún dispositivo de forma domótica. Con la ayuda de estos servicios de comunicación inalámbrica, se puede transferir voz, datos, videos, imágenes, etc.

Los Sistemas de Comunicación Inalámbrica también brindan diferentes servicios como videoconferencia, teléfono celular, buscapersonas, TV, Radio, internet de las cosas, etc. Debido a la necesidad de una variedad de servicios de comunicación, se desarrollan diferentes tipos de Sistemas de Comunicación Inalámbrica. Algunos de los importantes sistemas de comunicación inalámbrica disponibles en la actualidad son: [41]

- Televisión y Radiodifusión
- Comunicación por satélite
- Radar
- Sistema de teléfono móvil (comunicación celular)
- Sistema de posicionamiento global (GPS)
- Comunicación infrarroja

- WLAN (Wi-Fi)
- Bluetooth
- Teléfonos inalámbricos
- Identificación de radiofrecuencia (RFID)

### **Televisión y Radiodifusión**

Se considera que la radio es el primer servicio inalámbrico que se transmite. Es un ejemplo de un sistema de comunicación simplex donde la información se transmite solo en una dirección y todos los usuarios reciben los mismos datos.

### **Comunicación por satélite**

Las redes de comunicación satelital suministran cobertura mundial independiente de la densidad de población.

Los sistemas de comunicación satelital ofrecen telecomunicaciones (teléfonos satelitales), posicionamiento y navegación (GPS), radiodifusión, internet, etc. Otros servicios inalámbricos como el móvil, la televisión y otros sistemas de radio dependen de los sistemas de comunicación por satélite.

### **Sistema de teléfono móvil (comunicación celular)**

Tal vez, el sistema de comunicación inalámbrico más comúnmente utilizado es la tecnología de teléfono móvil. El desarrollo del dispositivo celular móvil cambió el mundo como ninguna otra tecnología. Los teléfonos móviles actuales no se limitan a hacer llamadas, sino que se integran con muchas otras funciones, como Bluetooth, Wi-Fi, GPS y radio FM.

### **Red de área local inalámbrica (WLAN)**

La red de área local inalámbrica o WLAN (Wi-Fi) es un servicio inalámbrico relacionado con Internet. Usando WLAN, diferentes dispositivos como computadoras portátiles y teléfonos móviles pueden conectarse a un punto de acceso y acceder a internet.

## Comunicación Bluetooth

Bluetooth es la especificación que se le da a las Redes Inalámbricas de Área Personal WPAN, y sirve para la transmisión de datos mediante una radiofrecuencia llamada banda ISM, aunque siempre en áreas cortas o personales. La distancia máxima que admite el sistema Bluetooth a la hora de intercambiar información con otro terminal, se clasifica en tres clases, como se muestra en la tabla 2.1. [42]

*Tabla 2.2: Clasificación del Bluetooth de acuerdo al alcance. [41]*

CLASE	ALCANCE
Clase 1	100 metros
Clase 2	25 metros
Clase 3	1 metro

## Características del Bluetooth

Las principales características de la comunicación Bluetooth son: [43]

- La tecnología inalámbrica Bluetooth está orientada a aplicaciones de voz y datos.
- Funciona en la banda de frecuencia de 2.4 GHz, que no precisa de ninguna licencia.
- Tiene un radio de acción de 10 o 100 metros dependiendo de la clase del dispositivo Bluetooth. La máxima velocidad de transmisión es de 3 Mbps.
- Los objetos sólidos no suponen ningún obstáculo para la tecnología inalámbrica Bluetooth.
- Tampoco es necesario que los dispositivos estén situados en la misma línea de visión, es decir, orientados uno frente a otro, ya que se transmite en todas direcciones.
- La seguridad siempre ha sido una de las prioridades en el desarrollo de la tecnología Bluetooth y continúa siéndolo. La especificación Bluetooth ofrece modos de seguridad acorde a las necesidades de los usuarios.

### **2.3 Propuesta de solución**

La implementación del prototipo, sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con Esclerosis Lateral Amiotrófica (ELA), permite reducir la frustración en los pacientes y familiares cercanos por problemas de comunicación presente en las personas con esta patología, a través del desarrollo del sistema electrónico se garantiza que la comunicación se establezca de forma eficiente, brindando todas las facilidades para que los pacientes puedan realizar sus actividades cotidianas.

## **CAPÍTULO 3**

### **METODOLOGÍA**

#### **3.1 Modalidad de la investigación**

La modalidad del proyecto, sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con Esclerosis Lateral Amiotrófica (ELA), se basa en investigación, desarrollo e implementación, la cual fue desarrollada mediante las siguientes técnicas:

Se utilizó la investigación aplicada, ya que se puso en práctica los conocimientos científicos adquiridos como también la información sobre la tecnología actual relacionada al tema, mismos que sirvieron para dar solución al problema planteado.

Se enmarcó dentro de la investigación bibliográfica debido a que el sustento científico del tema planteado se obtuvo de libros, publicaciones, artículos científicos y repositorios disponibles en el internet.

Se realizó la investigación de campo porque trataba de un estudio de los hechos, en el lugar que se produce, para ello se utilizó técnicas como la observación, la entrevista, que permitió estar en contacto directo al investigador con la finalidad de recolectar y registrar información.

#### **3.2 Población y muestra**

Por las características de la presente investigación. No se requiere población y muestra.

### **3.3 Recolección de la información**

La información se obtuvo principalmente de libros, revistas científicas, paper's y proyectos de titulación desarrollado en los últimos años, cada una de estas relacionadas y vinculadas a enfermedades de Esclerosis Lateral Amiotrófica (ELA), Sistemas de Comunicaciones Aumentativa y Alternativa (SAAC), Tecnologías de Comunicación y dispositivos de sistemas electrónicos, por lo que se requirió de la Investigación bibliográfica.

### **3.4 Procesamiento y análisis de datos**

Una vez seleccionada la información necesaria se llevó a cabo el siguiente procedimiento:

- Revisión crítica de la información recolectada.
- Interpretación de la información.
- Optimización de datos e información.
- Pruebas piloto.
- Control de errores.
- Presentación de resultados.

### **3.5 Desarrollo del proyecto**

A continuación, se presenta el desarrollo de las actividades necesarias que se efectuaron para el desarrollo del proyecto de investigación:

1. Análisis del movimiento de los párpados del ojo de las personas con ELA.
2. Estudio de los posibles métodos para adaptar a los ojos del paciente con ELA.
3. Selección del método adecuado para la etapa de sensorización ocular.
4. Desarrollo de la etapa de sensorización ocular.
5. Análisis de dispositivos electrónicos de ayuda para la comunicación de los pacientes con ELA.
6. Determinación óptima del dispositivo electrónico para la interface de texto, audio y control de dispositivos externos (etapa de visualización y control).
7. Desarrollo de la etapa de visualización y control.

- 8.** Estudio y selección de la comunicación inalámbrica a emplearse en el sistema electrónico.
- 9.** Calibración y adaptación de la etapa de sensorización ocular con la etapa de visualización y control.
- 10.** Implementación del prototipo sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con ELA.
- 11.** Pruebas, corrección de errores y validación del prototipo sistema electrónico planteado.

Elaboración del Informe Final.

## CAPÍTULO 4

### PROPUESTA

#### 4.1 Descripción general del Sistema

El desarrollo del proyecto de investigación, sistema electrónico para facilitar la comunicación y actividades cotidianas de los enfermos con ELA, se realizó de forma organizada con la finalidad de evitar fallos en la obtención de datos del paciente, realizar de manera eficiente y rápida el proceso de escritura y control de dispositivos electrónicos. En la Figura 4.1 se muestran todas las actividades de forma jerárquico que se emplearon para el desarrollo del sistema.

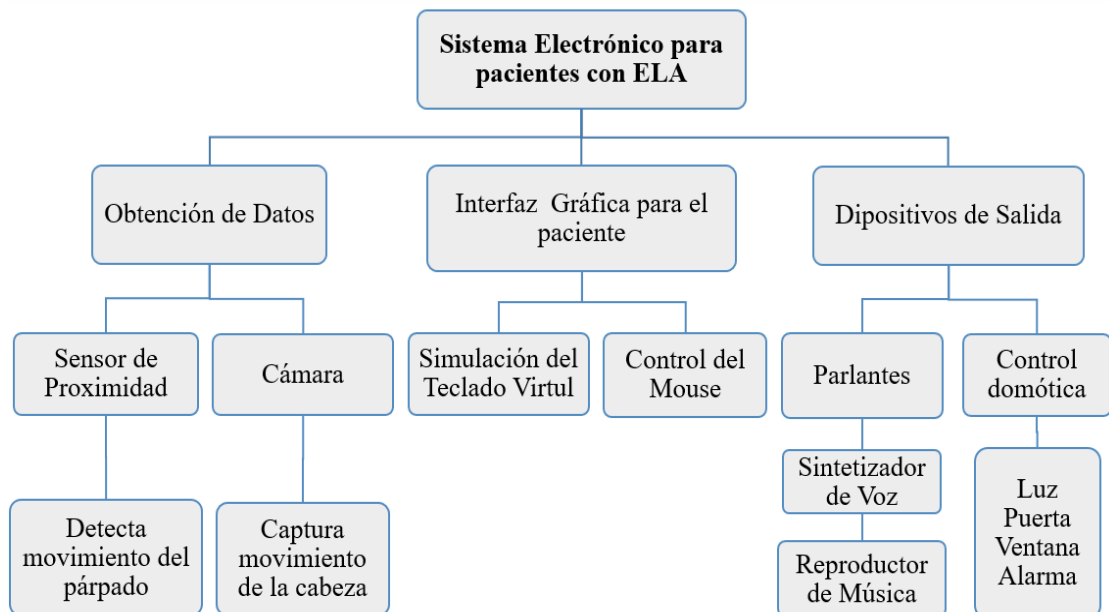


Figura 4.1: Organización jerárquica de actividades desarrolladas en el sistema

Elaborado por: Investigador



## 4.2 Análisis del movimiento de los párpados.

El movimiento de los párpados en una persona es de vital importancia, porque ayuda a mantener húmeda la córnea al momento de extender el líquido lacrimal, además protege el lóbulo ocular de lesiones o exceso de luz.

Las personas realizan el movimiento del párpado de acuerdo al estado de salud y la edad que tengan. En la Tabla 4.1 se muestra la velocidad y el número de parpadeos que comúnmente se efectúan.

*Tabla 4.1: Velocidad y número de parpadeos*

	<b>Movimiento de párpados en el tiempo</b>	<b>Velocidad del parpadeo</b>
Recién nacidos	1 o 2 parpadeos / min	200 – 300 ms
Adultos	10 a 15 parpadeos / min	300 – 400 ms
Al mirar fijamente un objeto	3 a 4 parpadeos / min	300 – 400 ms

*Elaborado por: Investigador*

## Movimiento de los párpados en pacientes con ELA

Las personas que padecen la enfermedad de ELA, en las etapas finales pierden el movimiento total de su cuerpo a excepción del movimiento ocular (párpados), este movimiento que realiza el paciente es algo lento, pero sigue siendo funcional, por esta razón se considera que el paciente puede aprovechar dicho movimiento para comunicarse con las personas de su alrededor e incluso realizar actividades cotidianas.

## 4.3 Análisis Técnico de Hardware

### 4.3.1 Sistema de Sensorización

Para detectar el movimiento voluntario del párpado o comisura lateral del ojo, se requiere de un sensor con excelente precisión, gran desempeño, tamaño reducido y otras características óptimas enfocadas a detectar movimientos de proximidad de forma eficiente y sobre todo sin afectar a los ojos del paciente. En la Tabla 4.2 se muestra el

análisis técnico de los diferentes sensores de proximidad infrarrojos que existen en el mercado.

*Tabla 4.2: Análisis técnico y selección del sensor de proximidad infrarrojo*

<b>Parámetros técnicos</b>	<b>Sensor óptico reflexivo CNY70</b>	<b>Sensor óptico reflexivo QRD1114</b>	<b>Módulo del Sensor óptico TCRT-5000</b>
Costo aproximado	2.50 \$	2 \$	4 \$
Longitud de onda	950 nm	940 nm	950 nm
Distancia de funcionamiento	0 - 5 mm	0 - 6 mm	0.2 - 15 mm
Filtro de luz solar dentro del sensor	Si	Si	Si
Tipo de censado	Sin contacto	Sin contacto	Con y sin contacto
Voltaje de funcionamiento	3.3 a 5 VCD	3.3 a 5 VCD	3.3 a 5 VCD
Corriente de trabajo	50 mA	20 mA	18 - 20 mA
Señal de Salida	No es digital	No es digital	Digital (0 o 1)
Comparador de Voltaje incluido	No	No	Si (LM393)
Ajuste de sensibilidad incluido	No	No	Si (Potenciómetro)
Dimensión	(LxWxH) 7x7x6 mm	(LxWxH) 6.1x4.6x4.9 mm	(LxWxH) 10.2x5.8x7 mm
Peso	1.2 g	1 g	3 g

*Elaborado por: Investigador*

Después del análisis de los diferentes sensores existentes en el mercado, se eligió el módulo del sensor óptico reflexivo TCRT5000, ya que es un sensor de proximidad de corto alcance, la luz infrarroja emitida por el sensor no afecta los ojos del paciente y gracias a su carcasa realizada en policarbonato de plástico con separadores, evita que se origine interferencias, además incluye un circuito integrado LM393 como comparador de voltaje para tener a la salida una señal digital que fácilmente puede ser interpretado por cualquier placa microcontrolador, posee un potenciómetro que permite la calibración a la distancia que desea detectar.

### **Características del sensor TCRT5000**

Las principales características del módulo del sensor óptico reflexivo TCRT5000 son las siguientes:

- Modelo del sensor: TCRT5000
- Voltaje operación: 3.5 a 5 Volts
- Corriente de operación: 20 mA
- Distancia de detección: 0.1 a 2.5 cm
- Distancia focal: 2.5 mm
- Circuito Integrado: LM393
- Dimensiones: 3.2 x 1.4 x 1.0 cm
- Peso: 3 g
- Salida: Digital (“1” y “0” lógico)
- Longitud de onda: 950 nm
- Incluye: Potenciómetro de calibración

### **4.3.2 Tarjeta controladora central**

Para facilitar la comunicación de los enfermos con ELA, es necesario que la interfaz gráfica del paciente sea grande, fácil de manipular y accesible. Por tal razón se realizó una comparativa entre miniordenadores SBC (Single Board Computer o Computadora de placa reducida), como se muestra en la Tabla 4.3. Se tomó en cuenta diferentes parámetros técnicos y de soporte, con la finalidad de que brinden al sistema las mejores prestaciones.

Tabla 4.3: Comparativa entre mini-ordenadores (SBC)

<b>Parámetros Técnicos</b>	<b>JaguarBoard</b>	<b>BeagleBone Black</b>	<b>Raspberry Pi 3</b>
Costo aproximado	\$ 130	\$ 110	\$ 70
CPU	Intel Atom Z3735G @ 1.83 Ghz.	ARM AM335x @ 1GHz	Quad-Core ARMv8 @ 1.2GHz de 64 bits.
RAM	1 GB DDR3.	512 MB	1 GB
Memoria interna	16 GB (ampliable vía micro-SD).	4 GB	No
Sistemas Operativos	Linux, Android, Windows	Debian, Android, Ubuntu, Cloud9	Raspbian, Ubuntu Mate, ArchLinux ARM, etc.
Conectividad	3 USB 2.0 HDMI 1.4 Ethernet	1 USB 2.0 Micro HDMI Ethernet	4 USB 2.0 HDMI 1.4 Ethernet WIFI Bluetooth 4.1 Bluetooth BLE
Puerto GPIO	4 Pines GPIO	69 pines GPIO	1x40 pines GPIO
Comunicación	I2C – Serial Port	I2C – SPI – Serial Port	UART – SDIO – SPI – I2C – I2S
Fuente de Alimentación	5V – 2 A	5V – 2 A	5V – 2 A
Compatible con pantalla de 10”	No	Si	Si

Elaborado por: Investigador

En relación al análisis técnico realizado, se eligió la tarjeta Raspberry Pi 3 de modelo B como el controlador central, debido a que suministra la posibilidad de conectar pantallas grandes de hasta 10" (pulgadas) o mediante el puerto de conexión y cable HDMI se puede conectar a la pantalla de TV o PC, además las características que posee hace que se comporte como un mini-ordenador al cual se puede adaptar varios componentes, tales como: cámara, teclado, mouse, USB, parlantes, y otros periféricos, además gracias a sus pines GPIO se comporta como un dispositivo o tarjeta para el control del IoT (internet de las cosas).

La tarjeta Raspberry Pi 3 de modelo B, se ajusta a las especificaciones requeridas para el desarrollo de nuestro sistema, estas especificaciones o requerimientos son las siguientes:

- Conexión a la pantalla Touch Screen mediante HDMI, para visualizar la interfaz gráfica.
- Puerto CSI de conexión a la cámara, para captar el rostro del paciente y controlar el cursor del mouse.
- Velocidad alta de procesamiento, para la adecuada ejecución de los algoritmos y/o programas.
- Puerto de conexión a los parlantes, para escuchar los mensajes escritos por el paciente.
- Módulo Bluetooth integrado, para controlar dispositivos externos mediante la comunicación Bluetooth.

Por las necesidades detalladas anteriormente, la tarjeta Raspberry Pi 3 resulto ser la indicada como el núcleo o controladora central del sistema, es más popular en comparación a las otras placas, posee mayor cantidad de información en los diferentes sitios web para realizar cualquier aplicación de forma sencilla, y además posee características más destacadas como Bluetooth 4.1 y Bluetooth BLE (de bajo consumo) integrado a diferencia de las otras placas que no tienen, estas y otras características se muestran en la Tabla 4.4.

### **Características de la Raspberry PI 3**

Las características generales que presenta la tarjeta Raspberry PI 3 se detallan a continuación:

Tabla 4.4: Características generales de la Raspberry Pi 3 modelo B

<b>ESPECIFICACIONES</b>	
CPU	Broadcom BCM2387 chipset. 1.2GHz Quad-Core ARM Cortex-A53 802.11 b / g / n LAN inalámbrica y Bluetooth 4.1 (Bluetooth Classic y LE)
GPU	Coprocesador multimedia Dual Core VideoCore IV®. Ofrece Open GL ES 2.0, acelerado por hardware OpenVG y 1080p30 H.264 de alto perfil de decodificación. Capaz de 1Gpixel / s, 1.5Gtexel / s o 24GFLOP con filtrado de textura e infraestructura DMA
Memoria	1GB LPDDR2
Sistema operativo	Boots desde la tarjeta Micro SD, ejecutando una versión del sistema operativo Linux o Windows 10 IoT.
Dimensión	85 x 56 x 17mm
Fuente	Conector Micro USB 5V, 2.5 A
<b>CONECTORES</b>	
Ethernet	Base de Ethernet 10/100 BaseT
Salida de Video	HDMI (rev 1,3 y 1,4 Compuesto RCA (PAL y NTSC)
Salida de Audio	Salida de audio de 3,5 mm, HDMI, USB 4 x conector USB 2.0
Pines GPIO	Cabezal de expansión de 40 pines de 2.54 mm (100 mil): tira de 2x20. Proporcionando 27 pines GPIO, así como líneas de suministro de +3.3 V, +5 V y GND
Conector de Cámara	Interfaz serial de la cámara MIPI de 15 pines (CSI-2)
Conector de Display	Display Serial Interface (DSI) Conector de cable plano flexible de 15 vías con dos líneas de datos y un carril de reloj.
Tarjeta de memoria	Tarjeta micro SD

Elaborado por: Investigador

### 4.3.3 Comparativa entre placas electrónicas microcontrolador

Para realizar la comunicación inalámbrica entre los lentes, la tarjeta RPI 3 y el pequeño sistema domótico, es necesario una placa de microcontrolador a la cual se pueda conectar un módulo Bluetooth y hacer una red de comunicación Bluetooth. En la Tabla 4.5 se muestra la comparativa entre las diferentes placas microcontroladores.

*Tabla 4.5: Comparativa entre diferentes placas electrónicas*

<b>Parámetros técnicos</b>	<b>Arduino</b>	<b>NODEMCU</b>	<b>Microcontrolador PIC</b>
Costo aproximado	\$ 20	\$ 13	\$ 6
Microcontrolador	ATmega 2560	RISC 32 bits MCU ESP 8266	PIC 16F628a
Pines E/S Digitales	54 y (6 son PWM)	16 pines GPIO	16
Pines de Entrada Analógica	16	1 de 10 bits (ADC)	4
Comunicaciones	SPI/I2C/UART	UART/SDIO/SPI/I2C	UART
Frecuencia de reloj	16 Hz	80 – 160 MHz	4 MHz (interno) 20 MHz (externo)
Memoria de Programa	256 Kb	96 Kb	2048 localizaciones de 14 bits
Memoria RAM	8 Kb	64 Kb	224 Bytes
Memoria Auxiliar EEPROM	4 Kb	-	128 Bytes
Voltaje de operación	5 V	3 a 3.6 V	5 V
Corriente de operación	40 – 50 mA	80 mA	25mA por Pin

*Elaborado por: Investigador*

Después del análisis de las diferentes placas electrónicas existentes se requiere la placa NODEMCU y el microcontrolador PIC 16F628A:

- La placa NODEMCU se utilizó para controlar mediante la comunicación Bluetooth el pequeño sistema domótico, se eligió esta placa porque es económico en relación al Arduino, además brinda la posibilidad de usar el IDE de Arduino como entorno de programación. Posee características adecuadas para el desarrollo del sistema entre ellas están:

#### **Características del NODEMCU**

- Procesador principal: ESP8266.
  - Protocolo inalámbrico 802.11 b/g/n.
  - Potencia de salida +19.5dBm en modo 802.11b.
  - Sensor de temperatura integrado.
  - Corriente en espera: < 10uA.
  - Comunicación UART, SPI, I2C.
- El microcontrolador PIC 16F628A se usó para la comunicación Bluetooth entre los lentes y la interfaz gráfica del paciente, principalmente se eligió este microcontrolador por su tamaño reducido la cual permitió adaptar de forma sencilla a los lentes que usara el paciente, además el módulo del sensor TCRT5000 es compatible con el PIC mientras que con la placa NODEMCU no funciona correctamente debido a que sus pines de entrada trabajan con solo 3.3V, estos aspectos fueron esenciales para la elección adecuada del microcontrolador PIC, sus principales características se detallan a continuación.

#### **Características del microcontrolador PIC 16F628A**

- Memoria de datos RAM: 36 bytes (PIC16C84) y 68 bytes (PIC16F84).
- Memoria de datos EEPROM: 64 bytes.
- Interrupciones: 4 tipos diferentes.
- Frecuencia de trabajo: 10MHz máxima.
- Líneas de E/S Digitales: 13 (5 Puerta A y 8 Puerta B).
- Corriente máxima absorbida: 80mA Puerto A y 150mA Puerto B.
- Corriente máxima suministrada: 50mA Puerto A y 100mA Puerto B.



- Corriente máxima absorbida por Pin: 25mA.
- Corriente máxima suministrada por Pin: 20mA.
- Voltaje de alimentación (VDD): De 2 a 6 V DC.
- Voltaje de grabación (VPP): De 12 a 14 VDC.

#### 4.3.4 Comparativa entre pantallas Touch Screen

La pantalla Touch Screen es una pantalla táctil que mediante un toque directo sobre la superficie permite ingresar datos, de esta manera ordena al dispositivo para que realice diferentes acciones. En nuestro caso se usó la pantalla touch screen de 10.1” para visualizar la interfaz gráfica que es manejado por el paciente. También la pantalla puede ser manipulada por la persona que está a cargo del paciente. En la Tabla 4.6 se muestra las diferentes pantallas compatibles con la placa Raspberry Pi 3.

*Tabla 4.6: Comparativa de las pantallas para la RPI 3*

<b>Parámetros técnicos</b>	<b>Pantalla de 5 ”</b>	<b>Pantalla de 7 ”</b>	<b>Pantalla de 10.1 ”</b>
Costo aproximado	35 \$	50 \$	106 \$
Resolución	800 x 480 HD	1024 x 600	1280 x 800
Sistema Operativo	Soporta todo relacionado a RPI	Only Rasbian y ubuntu	Soporta todo relacionado a RPI
Pantalla táctil	SPI via RPI	IPS screen, full view	IPS screen, full view
Dimensión	120 mm x 74 mm x 7 mm	210 mm x 120 mm x 8 mm	265 mm x 187 mm x 10 mm
Peso sin paquete	125 g	190 g	280 g
Peso con paquete	145 g	220 g	310 g

*Elaborado por: Investigador*

Después del análisis técnico realizado se eligió la pantalla Touch Screen de 10.1” para presentar la interfaz gráfica al paciente. Esta pantalla tiene la resolución y dimensión adecuada para visualizar de mejor manera la simulación del teclado virtual, de tal forma que el paciente podrá acceder y controlar con facilidad todas las funcionalidades del sistema. Además, como la pantalla es táctil brinda la posibilidad de que manipulen personas que cuidan al paciente.

### **Características de la pantalla Touch Screen 10.1”**

#### LCD y placa principal

- Pantalla IPS con una resolución de  $1280 \times 800$ .
- LCD capacitiva, admite 10 puntos de contacto.
- Soporta sistemas operativos como: Raspbian, Ubuntu, Windows 10 IoT. Sin unidad con un solo toque.
- Admite BB Black, proporciona Angstrom y reflejo relacionado.
- La interfaz HDMI se usa para la visualización mientras que la interfaz USB se usa para la Touch.
- Soporta Backlight Control

#### Soporte Acrílico

- Adopta los medidores de alta calidad de acrílico negro.
- Inclinación de la pantalla:  $30^\circ$  o  $45^\circ$
- Orificios de montaje reservados para placa externa. Compatible con Raspberry Pi 3B / 2B / B + / A + / B, BB Black y Banana Pi.

### **4.3.5 Parlantes**

Para poder reproducir los mensajes escritos por el paciente y escuchar música, es necesario utilizar un altavoz, para ello se seleccionó el parlante de computadora ya que tiene un conector de entrada USB que permite la alimentación para que el parlante funcione, se puede escuchar el sonido vía auricular la cual tiene una entrada de tipo Plug compatible con el conector RCA de la tarjeta RPI 3. Además, su tamaño es ideal, fácil de manipular y su costo es bajo. A continuación, se presenta las características más importantes del parlante y seguidamente en la Figura 4.2 se muestra los parlantes seleccionados.

### **Características del parlante de computadora**

- Conector de entrada USB
- Entrada tipo Plug (Auriculares)
- Control para volumen
- Excelente salida de sonido
- Surtido en colores
- Bajo costo



*Figura 4.2: Parlantes para conectar a la placa RPI 3.*

### **4.3.6 Comparación de Tecnología inalámbrica**

La comunicación inalámbrica es un gran avance tecnológico que evita la molestia del uso de cables, además ofrece la facilidad de acceder a la red desde cualquier lugar a través de ondas electromagnéticas. En nuestro caso es de vital importancia la comunicación inalámbrica para el control de dispositivos externos y la interacción con la interfaz gráfica, debido a que el paciente no puede realizar ninguna actividad al estar casi completamente paralizado y necesita de ayuda alterna para realizar dichas actividades. Esta comunicación debe ser segura evitando completamente el riesgo de que alguna persona pueda acceder a la red, además debe tener bajas tasas de transmisión de datos y bajo consumo de energía. Por tal razón, en la Tabla 4.7 se presenta una comparativa de la comunicación inalámbrica óptima y que se adapte a las necesidades de nuestro sistema.

Tabla 4.7: Comparativa de comunicaciones inalámbricas

	<b>WIFI</b>	<b>Zigbee</b>	<b>Bluetooth</b>
Descripción	WiFi significa fidelidad inalámbrica.	Comunicación inalámbrica similar al Bluetooth pero con mayor alcance.	Es una comunicación inalámbrica de corto alcance.
Arquitectura	WLAN (Red Inalámbrica de Área local)	WPAN (Red Inalámbrica de Área Personal)	WPAN (Red Inalámbrica de Área Personal)
Estándar	IEEE 802.11	IEEE 802.15.4	IEEE 802.15.1
Velocidad	Hasta 300 Mbps	Hasta 250 Kbps	3 Mbps
Frecuencia	(2,4 – 5) GHz	868 MHz (Europa) 900-9228 MHz (NA) 2,4 GHz (Todo el mundo)	2,4 GHz
Alcance	Hasta 100 metros	Hasta 100 metros	Hasta 30 metros
Seguridad	Moderada	Baja	Baja
Consumo energético	Elevado	Muy Bajo	Bajo
Integrado en la RPI 3	SI	NO	SI

*Elaborado por: Investigador*

Después de la comparativa de la comunicación inalámbrica más usada a nivel de área local (WLAN) y área personal (WPAN), se optó por la comunicación Bluetooth porque cuenta con características y ventajas apropiadas para el diseño de nuestra red inalámbrica, es de bajo consumo energético, el alcance esta hasta los 30 metros, distancia suficiente para el control de cualquier dispositivo. Además, la tarjeta RPI 3 tiene integrado Bluetooth 4.1, necesitando de solo un módulo externo (HC-05) adicionalmente para realizar la conexión, configuración y el uso de manera eficiente y confiable. A continuación, se detallan algunas características importantes del módulo Bluetooth:

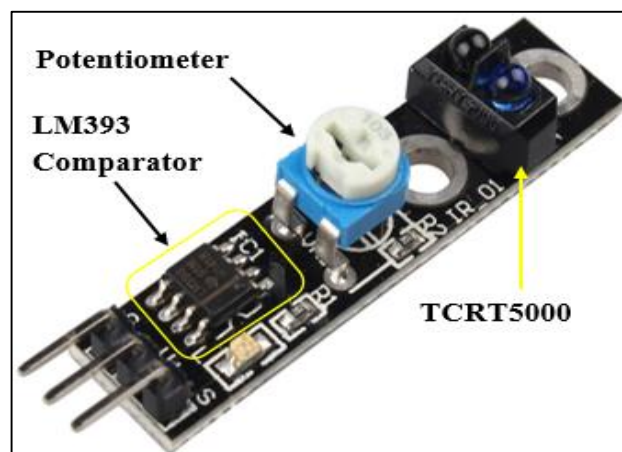
- No necesita línea de Vista
- Voltaje de alimentación: 3.3VDC – 6VDC.
- Voltaje de operación: 3.3VDC.
- Baud rate ajustable: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
- Corriente de operación: < 40 mA
- Corrige errores en la transmisión porque es un Protocolo.

#### 4.4 Obtención de datos

##### 4.4.1 Detector de Proximidad

##### Módulo del sensor TCRT5000

El sensor TCRT5000 está colocado en los lentes del paciente frente al ojo. El globo ocular y el párpado reflejan la luz IR de manera diferente, si el paciente realiza un movimiento voluntario del párpado, el sensor produce un voltaje que es comparado por el LM393 para obtener a la salida un voltaje digital, esta señal ingresa al PIC 16F628A como una interrupción externa, la cual es procesada y transmitida mediante la comunicación Bluetooth a la tarjeta RPI 3 para realizar el click izquierdo del mouse que se desplaza en el teclado virtual. El potenciómetro que incluye el módulo permite calibrar la precisión del sensor para detectar de mejor manera el movimiento del párpado. La luz infrarroja que emite el sensor no afecta el glóbulo ocular del paciente. En la Figura 4.3 se presenta el módulo del sensor TCRT5000.



*Figura 4.3: Módulo del sensor TCRT5000*

*Elaborado por: Investigador*

El módulo TCRT5000 evito realizar un circuito adicional donde necesariamente debíamos hacer el acondicionamiento de señal con el potenciómetro y un comparador para tener a la salida una señal digital pura. Este módulo tiene incluido dicho potenciómetro y comparador que permitió trabajar con más facilidad y confianza. En la Figura 4.4 se muestra el circuito eléctrico de módulo TCRT5000.

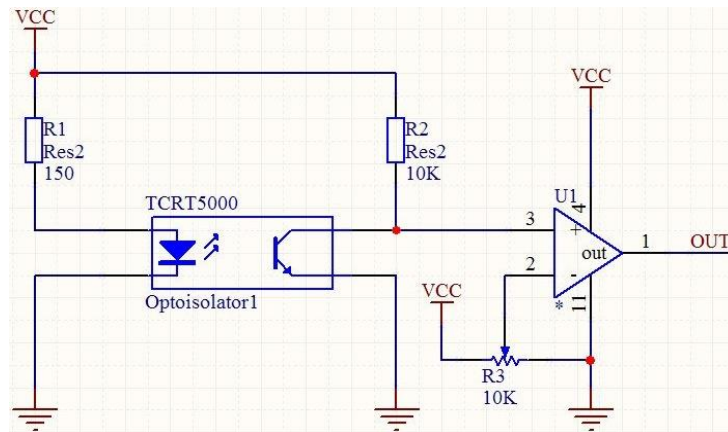


Figura 4.4: Circuito Eléctrico del módulo TCRT5000

Elaborado por: Investigador

### Ubicación del Sensor en los lentes

Inicialmente, se ubicó el sensor en los lentes con el propósito de calibrar y comprobar el adecuado funcionamiento del módulo TCRT5000. En la Figura 4.4, se puede observar que el sensor se encuentra ubicado en los lentes, más adelante se detallara el circuito final que se realizó para la etapa de los lentes.



Figura 4.5: Ubicación del sensor en los lentes

Elaborado por: Investigador

## Calibración del sensor

Para realizar la calibración se tomó en cuenta las características eléctricas que presenta el sensor, en una de ellas menciona que la salida es de señal digital y que la distancia de detección se puede variar con el potenciómetro. Para ello se realizó un pequeño programa que permite leer datos del sensor y cuando detecte movimientos calibrar la precisión variando el potenciómetro. En la Tabla 4.8 se muestra los valores que se obtuvieron en la calibración del sensor.

Tabla 4.8: Calibración del sensor

Acción del sensor	Salida	Voltaje medido	Efectividad del sensor
Potenciómetro a 180 °	1.3 cm	0.47 V	Regular
Potenciómetro a 160 °	0.9 cm	0.35 V	Bueno
Potenciómetro a 130 °	0.5 cm	0.24 V	Excelente
Potenciómetro < a 100 °	0.3 cm	0.22 V	Regular
Si detecta movimiento	“0” lógico	0.24 V	-
No detecta movimiento	“1” lógico	4.7 V	-

*Elaborado por: Investigador*

## Diagrama eléctrico empleado en los lentes del paciente

El diagrama eléctrico que se presenta en la Figura 4.6, consta de una batería lipo de 7.4 Vdc, la cual es estabilizado con el regulador de voltaje LM7805 para obtener a la salida una tensión constante de 5 VDC. Este voltaje ideal para que funcione el microcontrolador PIC 16F628A, el módulo del sensor TCRT5000 y el módulo Bluetooth HC-05.

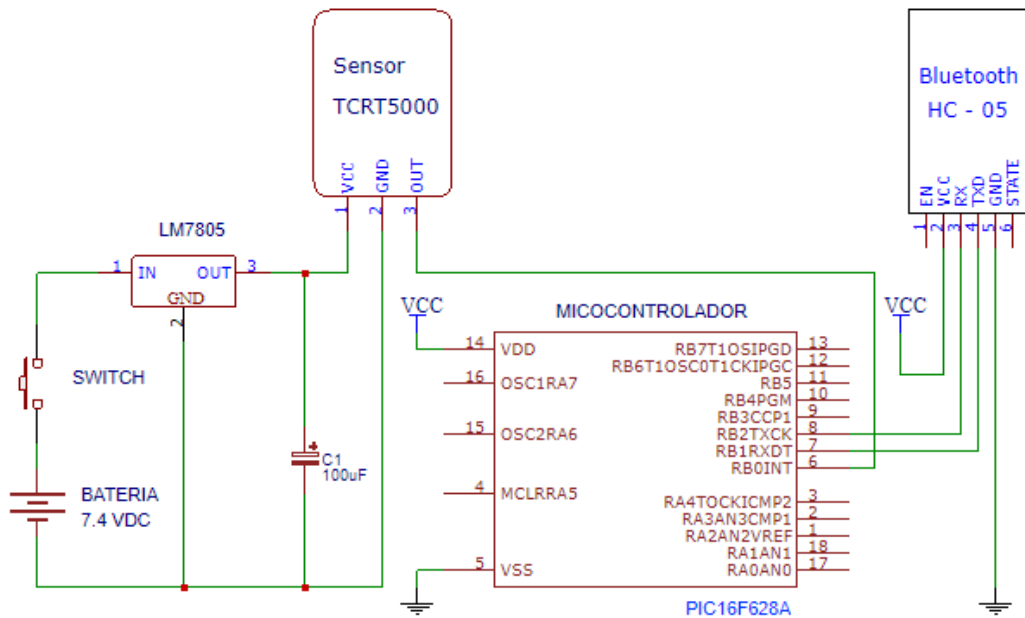


Figura 4.6: Diagrama eléctrico para la etapa de lentes

Elaborado por: Investigador

### Análisis de consumo energético

Para que el circuito funcione es necesario de una fuente de alimentación o batería, para alimentar nuestro prototipo se utilizó la batería lipo de 7.4 VDC con una corriente de 300 mAh, se eligió esta batería por su tamaño reducido y menor peso, es ideal en aplicaciones sumamente pequeñas, como es el caso de nuestro prototipo, puesto que debe ser liviano y pequeño para adaptar con facilidad a los lentes. El cálculo de la vida útil o el tiempo que se demora la batería en descargar, se lo puede realizar utilizando la siguiente ecuación (1) de la potencia eléctrica.

$$P = V * I \quad (1)$$

Donde:

P = Potencia en vatios (W)

V = Voltaje en voltios (V)

I = Corriente en amperios (A)

#### a) Capacidad de la batería

Es necesario conocer cuántos vatios hora (Wh) almacena la batería, para ello se realizó el simple cálculo de la potencia eléctrica utilizando la ecuación (1), aunque en algunas



baterías este dato viene en sus características eléctricas y no es necesario realizar este cálculo:

$$Potencia_{Bateria} = V * I$$

$$Potencia_{Bateria} = 7.4 V * 300mAh$$

$$Potencia_{Bateria} = 2.22 Wh \text{ (watts hora)}$$

### b) Potencia total del circuito

Para poder realizar el cálculo de la potencia total del circuito, se revisó las características eléctricas de voltaje y corriente de cada uno de los dispositivos, posteriormente se aplicó la fórmula de la potencia eléctrica ( $P = V * I$ ) descrita anteriormente y se realizó la Tabla 4.9, donde se muestra la potencia que suministra cada elemento y la potencia total que consume nuestro circuito.

Tabla 4.9: Potencia total que consume el circuito de los lentes

Dispositivo	Voltaje	Corriente	Potencia
PIC 16F628A	5 V	80 mA	0.4 W
Sensor TCRT5000	5 V	20 mA	0.1 W
Bluetooth HC-05	5 V	40 mA	0.2 W
<b>Total</b>			0.7 W

*Elaborado por: Investigador*

### c) Tiempo de descarga de la batería

Una vez obtenido el valor de la potencia de la batería y la potencia total que consume el circuito, se realizó el cálculo del tiempo que se demora la batería en descargarse completamente, para ello se utilizó la siguiente ecuación (2):

$$TiempoDescarga_{Bateria} = \frac{Potencia_{Bateria}}{Potencia_{Total_{Circuito}}} \quad (2)$$

$$TiempoDescarga_{Bateria} = \frac{2.22 \text{ Wh}}{0.7 \text{ W}}$$

$$TiempoDescarga_{Bateria} = 3.17 \text{ h (horas)}$$

La batería se descargará completamente en un tiempo de 3 horas aproximadamente, el paciente podrá realizar actividades externas o comunicarse sin inconvenientes durante ese lapso de tiempo, si la batería se descarga la persona que asiste al paciente la cargará y se volverá usar el prototipo sin ningún problema.

Si desea utilizar el prototipo de los lentes por más tiempo entonces se necesita una batería con mayor capacidad, a continuación, se realizará un análisis para la batería de mayor capacidad.

#### **Análisis de batería con capacidad de 2200 mAh**

Para el análisis del tiempo que se demora en descargar la batería con 7.4 V y capacidad de 2200 mAh, se realizó los mismos cálculos presentados anteriormente:

##### **a) Capacidad de la batería**

Para calcular la potencia de la batería se utilizó la fórmula de la potencia eléctrica como muestra en la siguiente ecuación:

$$Potencia_{Bateria} = V * I$$

$$Potencia_{Bateria} = 7.4 \text{ V} * 2200 \text{mAh}$$

$$Potencia_{Bateria} = 16.28 \text{ Wh (watts hora)}$$

##### **b) Potencia total del circuito**

La potencia total que consume el circuito de los lentes no varía, por lo tanto, se tiene 0.7 W (watts) de consumo.

### c) Tiempo de descarga

Finalmente, como ya se conoce la capacidad de la batería y la potencia total del circuito, entonces se procedió a realizar el cálculo del tiempo de descarga de la batería, usando la siguiente ecuación:

$$TiempoDescarga_{Bateria} = \frac{Potencia_{Bateria}}{Potencia Total_{Circuito}} \quad (2)$$

$$TiempoDescarga_{Bateria} = \frac{16.28 Wh}{0.7 W}$$

$$TiempoDescarga_{Bateria} = 23.25 h \text{ (horas)}$$

Esta batería ofrece una durabilidad de 23 horas aproximadamente, pero se debe tomar en cuenta que la batería que proporciona mayor capacidad también tiene mayor peso. En este caso el peso de la batería de 2200 mAh está alrededor de 139 gramos, mientras que la batería de 300 mAh tiene un peso mucho menor que es de 25 gramos, para realizar nuestro prototipo se utilizó la batería con menor peso.

### Implementación del prototipo para los lentes

Antes de desarrollar el prototipo final de los lentes, se realizó la respectiva simulación y se verificó el funcionamiento en la protoboard, las pruebas realizadas inicialmente fueron defectuosas debido a que el sensor debía estar ubicado de manera adecuada, pero este error se logró solucionar calibrando la distancia de detección del sensor, después que el sensor esté funcionando efectivamente, se procedió a enviar el dato hacia la tarjeta RPI 3 a través del módulo Bluetooth HC-05, y los resultados fueron satisfactorios.

Finalmente se implementó el prototipo que tienen una dimensión de 7 cm de ancho, 3 cm de alto y una profundidad de 2.5 cm. El peso es de 150 gramos aproximadamente, se puede adaptar de manera sencilla ya que tiene una pequeña abertura en la parte lateral derecha para ubicar o retirar de los lentes sin ningún problema. Además, tiene un switch que permite manipular el encendido o apagado del circuito con la finalidad de prolongar la vida útil de la batería empleada. En la Figura 4.7 se muestra el prototipo final colocado en los lentes.



*Figura 4.7: Prototipo final de los lentes*

*Elaborado por: Investigador*

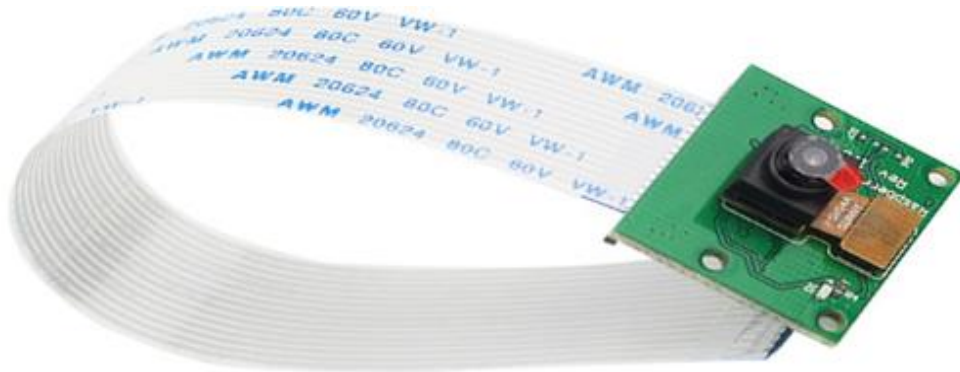
#### **4.4.2 Detección del Movimiento de Cabeza**

Los pacientes que son afectados por ELA, en la etapa tardía, pueden mover levemente alguna parte de su cuerpo incluyendo la cabeza. Sin embargo, la mayoría de estos pacientes, pierden movilidad de la lengua, ocasionando pérdida del habla y como consecuencia impidiendo la comunicación con las personas cercanas. Para ayudar a este grupo de pacientes surgió la necesidad de buscar varias alternativas de comunicación, finalmente se encontró la opción adecuada, que consiste en usar el módulo de la cámara RPI para que detecte el movimiento de la cabeza.

#### **Módulo de la cámara RPI**

El módulo de la cámara Raspberry Pi, como se muestra en la Figura 4.8, es muy especial que captura fotos fijas y graba videos de alta definición, es económico, fácil de utilizar y presenta características avanzadas para realizar múltiples aplicaciones.

En nuestro caso la cámara RPI es una alternativa de comunicación para pacientes con ELA que pueden mover su cabeza. Consiste en capturar y transmitir hacia la placa electrónica RPI 3 el movimiento hacia arriba, abajo, izquierda, derecha, oblicua que realiza el paciente para controlar el cursor del mouse.



*Figura 4.8: Módulo de cámara RPI. [44]*

#### **Características que debe cumplir la cámara RPI**

- Los fotogramas por segundo (fps) deben ser superiores a 25 para que la transmisión sea correcta.
- El lente de la cámara debe ser de cristal.
- El sensor de la cámara debe ser CCD (Charged Coupled Device- Dispositivo Acoplado Cargado).
- Debe tener una base adaptable a la superficie.

#### **4.5 Análisis de Software**

Para la administración de la información, datos y realizar determinadas tareas es necesario realizar un análisis de los diferentes tipos de software, para lo cual a continuación se detalla el análisis comparativo de los distintos programas existentes en el mercado.

#### **Comparación de los sistemas operativos**

Existen varios sistemas operativos que pueden ser privados (pagados) o libres (gratis), los cuales se encargan de gestionar todos los recursos del computador, de esta forma permite la comunicación entre el usuario y el ordenador. Por esta razón es necesario realizar una comparativa en donde se puede identificar las características de los sistemas

operativos más comunes y seleccionar la apropiada para el desarrollo del sistema. En la Tabla 4.10 se muestra los sistemas operativos más reconocidos.

*Tabla 4.10: Sistemas operativos más comunes.*

	<b>Windows</b>	<b>Mac OS</b>	<b>Linux</b>
Descripción	Es un sistema operativo desarrollado por la empresa Microsoft Corporation, el cual se encuentra dotado de una interfaz gráfica de usuario basada en el prototipo de ventanas.	Es un tipo de sistema operativo exclusivo desarrollado para los terminales creados por la compañía Apple y significa Macintosh Operating System (Mac OS)	Linux es un sistema operativo de software libre, no es propiedad de ninguna empresa por ende no es necesario comprar una licencia para instalar y usar en un equipo informático.
Ventajas	Es el S.O más reconocido a nivel mundial, posee la mayor cantidad de software desarrollado y es muy fácil de utilizar.	El sistema operativo es ideal para personas que trabajan como diseñadores ya que cuenta con la mejor interfaz gráfica que existe en el mercado.	Posee una gran cantidad de software libre para utilizar con Linux. Es usado en servidores de rendimiento alto, con una gran estabilidad.
Desventajas	El costo es muy alto, las nuevas versiones requieren muchos recursos, pueden ser vulneradas con facilidad y es muy inestable	Existe poco Software desarrollados para este sistema, el costo es elevado y si se daña es difícil dar soluciones rápidas.	No existe mucho software comercial, se debe saber UNIX para poder usar el sistema.
Acceso	Privado	Privado	Libre

*Elaborado por: Investigador*

Después del análisis de los sistemas operativos más populares, se elegido Linux ya que es un software libre y aunque no tiene variedad de softwares desarrolladas actualmente la comunidad Linux está trabajando para brindar mayores y mejores softwares para el

futuro. Además, es muy estable y no necesita de licencia para poder utilizar. Algunas de las características importantes de Linux se detallan a continuación:

- Sistema operativo totalmente libre.
- Multitarea: se puede ejecutar al mismo tiempo numerosos programas.
- Multiusuario: más de una sola persona puede trabajar en la misma versión de la misma aplicación de una manera simultánea.
- Redes y telecomunicaciones: al soportar varias interfaces de red brinda mayores prestaciones.
- Conexión a internet, interconectividad, programación con varias herramientas, portabilidad, ambiente gráfico, etc.

### Distribuciones de Linux para Raspberry Pi

La Raspberry Pi ha cosechado grandes éxitos por su bajo precio y por sus grandes posibilidades, tanto para los desarrolladores de software, estudiantes de informática, como para los aficionados que emplean para sus propios proyectos. El sistema operativo oficial es el Raspbian. Sin embargo, existen otros sistemas operativos que se pueden usar de acuerdo a las necesidades del usuario. En la Tabla 4.11 se presenta una comparativa de los sistemas operativos que más se emplean en la tarjeta Raspberry Pi.

*Tabla 4.11: Sistemas operativos para la tarjeta Raspberry Pi.*

	<b>Raspbian</b>	<b>Archlinux ARM</b>	<b>Ubuntu Mate</b>
<b>Descripción</b>	Es el sistema Operativo oficial de la Raspberry Pi	Es un puerto de Arch Linux para procesadores ARM. Su filosofía de diseño es simplicidad y control total para el usuario final.	Ubuntu MATE es una distribución Linux basada en Ubuntu.
<b>Distribución de Linux</b>	Debian Wheezy	Minimalista de Linux	Ubuntu

<b>Modelo de Desarrollo</b>	Software libre y de código abierto	Software libre y de código abierto	Software libre y de código abierto
<b>Lanzamiento inicial</b>	Junio del 2012	Marzo del 2002	Octubre 2014
<b>Última versión estable</b>	Junio del 2017	Mayo 2017	Agosto 2017
<b>Núcleo</b>	Linux	Linux	Linux
<b>Tipo de núcleo</b>	Monolítico	Linux	Monolítico
<b>Plataformas soportadas</b>	ARMv6	x86, IA-32 y x86-64, CPU ARM de gama baja y alta.	x86, x86-64, PowerPC, ARMv7
<b>Sistema de gestión de paquetes</b>	dpkg	dpkg	Dpkg
<b>Licencias</b>	GPL y otras libres	GPLv2 y otros	GPL y otras
<b>Interfaz gráfica por defecto</b>	PIXEL	KDE GNU/Linux	MATE
<b>Herramientas de Desarrollo</b>	IDLE para Python	IDLE para Python	IDLE para Python
<b>Soporta Eviacam</b>	No	No	Si

*Elaborado por: Investigador*

Al finalizar el análisis de los diferentes sistemas operativos que se pueden instalar en la tarjeta Raspberry Pi 3, se optó por Ubuntu Mate, porque presenta un entorno de escritorio amigable y fácil de usar, pero la ventaja más importante para elegir este sistema operativo fue que permitió instalar el Software de EVIACAM para poder usar



la cámara de la RPI. Raspbian es el sistema operativo popular pero no es compatible con el software mencionado anteriormente, por tal razón se descartó al sistema operativo más reconocido para la placa RPI y se eligió Ubuntu Mate que es la distribución de Linux basado en Ubuntu y reconocido por Canonical.

Algunas de las características de Linux que emplea el sistema operativo Ubuntu Mate son las siguientes:

- Linux kernel 4.19.1
- MATE 1.12.2
- Firefox 46.0.1
- Thunderbird 38.8.0
- LibreOffice 5.1.2.2
- Shotwell 0.22.0
- Rhythmbox 3.3
- VLC media player 2.2.2
- Scratch
- Onboard
- Minecraft Pi
- Python (2.7 & 3.5)
- IDLE (Integrated Development Environment for Python)
- Tilda (F12 drop-down terminal emulator)

#### **4.6 Diseño del sistema Propuesto**

El sistema fue desarrollado para detectar el movimiento del párpado a través del sensor de proximidad y el movimiento de la cabeza utilizando el módulo de cámara RPI. En la primera opción el paciente con ELA realiza un movimiento voluntario del guiño o párpado del ojo, el sensor de proximidad ubicado en los lentes capta y entrega esta señal al microcontrolador PIC la cual envía el dato a la placa RPI 3 mediante la comunicación Bluetooth, el algoritmo desarrollado en Python toma este dato como una interrupción para realizar el click izquierdo del mouse que se desplaza sobre el teclado virtual, con ello se puede escribir el mensaje deseado, acceder a una sub-ventana donde se encuentran mensajes predefinidos con las necesidades básicas del paciente, reproducir

estos mensajes e incluso escuchar música utilizando los parlantes, además se puede acceder satisfactoriamente a las funciones del pequeño sistema domótico para controlar el estado de la luz, abrir o cerrar la puerta y activar o desactivar la alarma, todo esto a través de la comunicación Bluetooth. En la segunda opción el módulo de la cámara RPI captura el movimiento de la cabeza que realiza el paciente y controla el cursor del mouse, para realizar el proceso anterior con la diferencia que el tiempo de ejecución es más rápido. En la Figura 4.9 se muestra el esquema general del sistema.

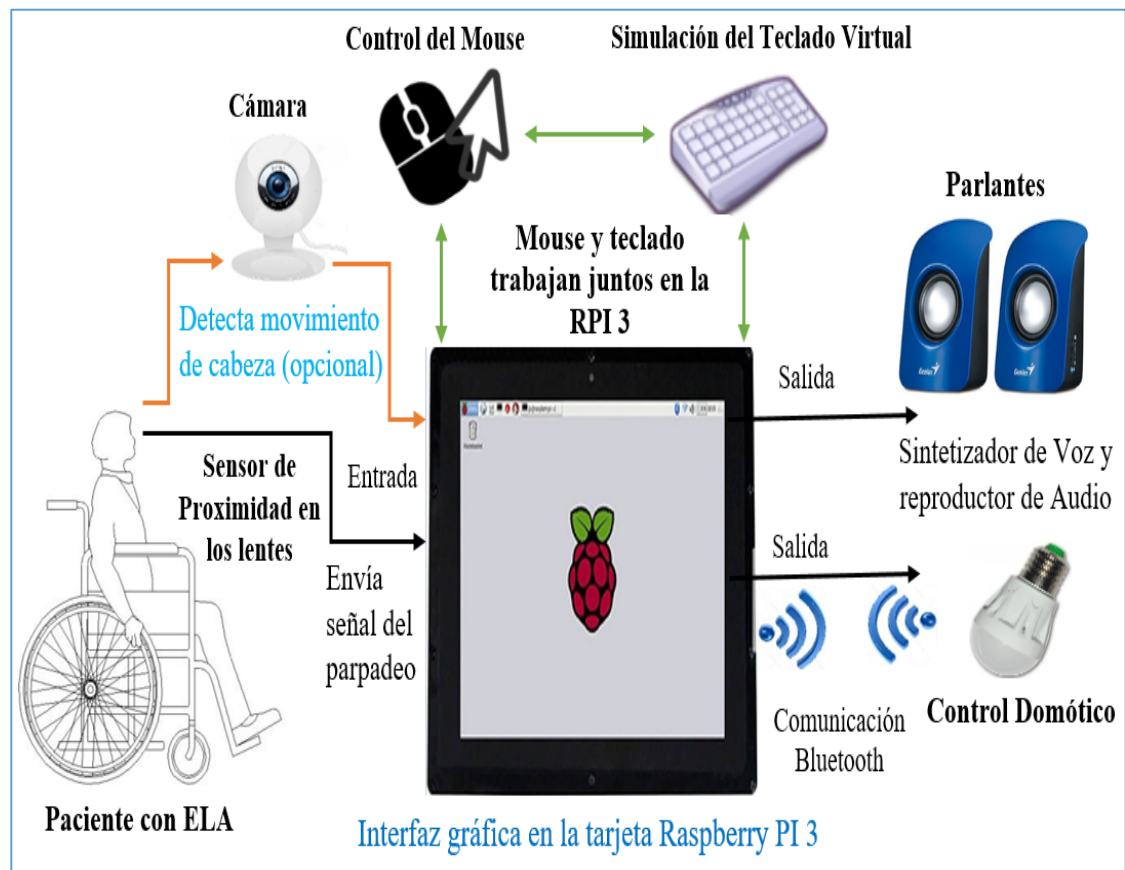


Figura 4.9: Esquema general del sistema

Elaborado por: Investigador

#### 4.7 Diagrama general del sistema programado

El sistema electrónico para ayudar a pacientes con ELA, está representado mediante un diagrama de flujo general como se muestra a continuación en la Figura 4.10.

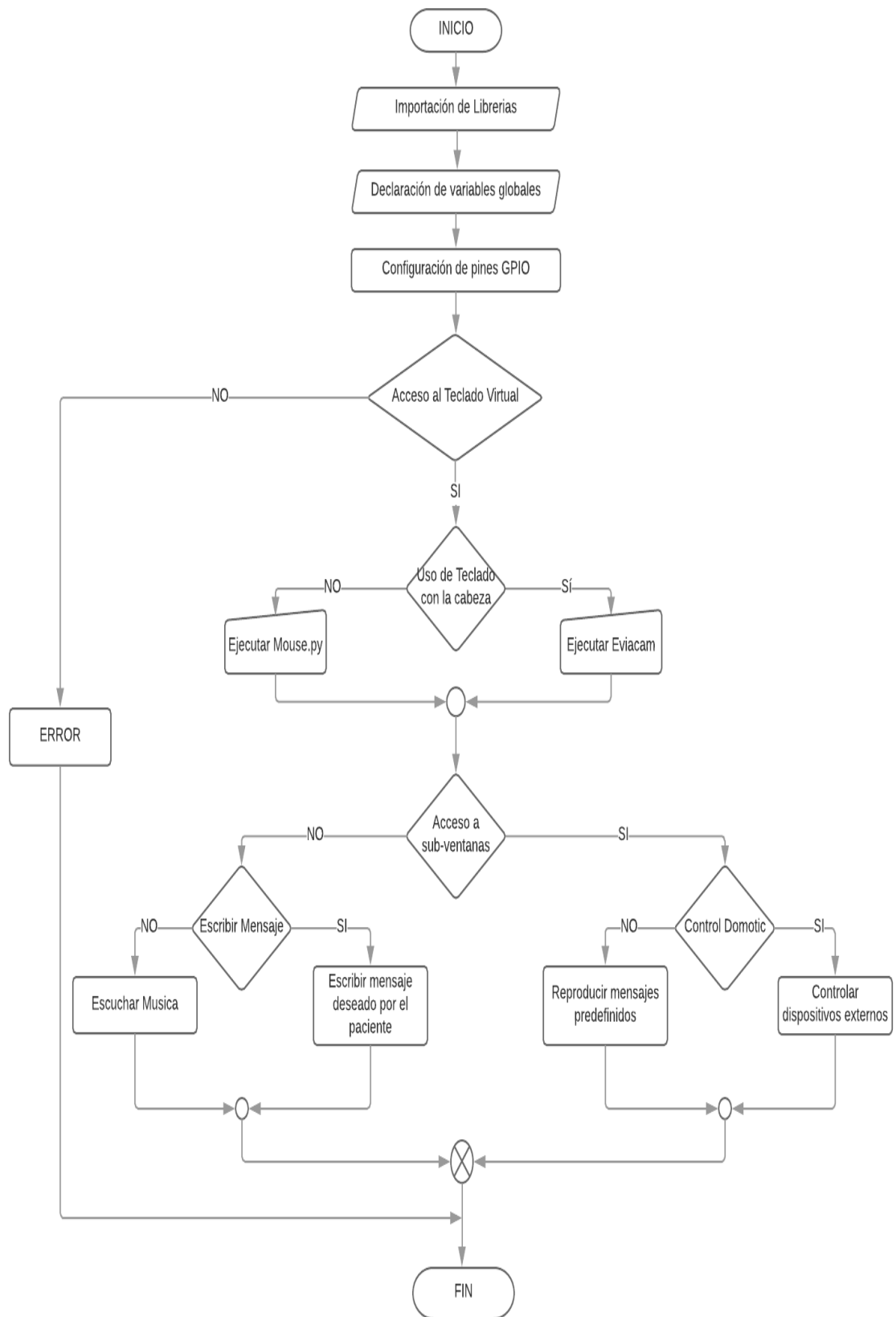


Figura 4.10: Diagrama de flujo del funcionamiento del sistema

Elaborado por: Investigador

## 4.8 Desarrollo del Sistema

### 4.8.1 Instalación de Ubuntu Mate en RPI 3

Para instalar Ubuntu Mate 16.04 LTS en la tarjeta Raspberry Pi 3, se utilizó la micro SD de 16 GB (núcleo 10), como no existe un disco duro en la Raspberry Pi entonces la SD cumple con esta función, por tal razón es muy importante que la micro SD tenga características óptimas para su adecuado funcionamiento. Previamente se realizó los siguientes pasos:

- Se descargó la imagen de Ubuntu Mate desde la página oficial de la Raspberry Pi.
- Se descargó e instaló en Windows la aplicación de Win32 Disk Imager para poder cargar la imagen al micro SD.

#### a) Cargar la imagen de Ubuntu Mate al micro SD.

Para cargar la imagen al micro SD se ejecutó Win32 y se formateó la tarjeta, después buscamos la ruta de la imagen, cargamos, escribimos y después expulsamos, como se muestra a continuación en la Figura 4.11.

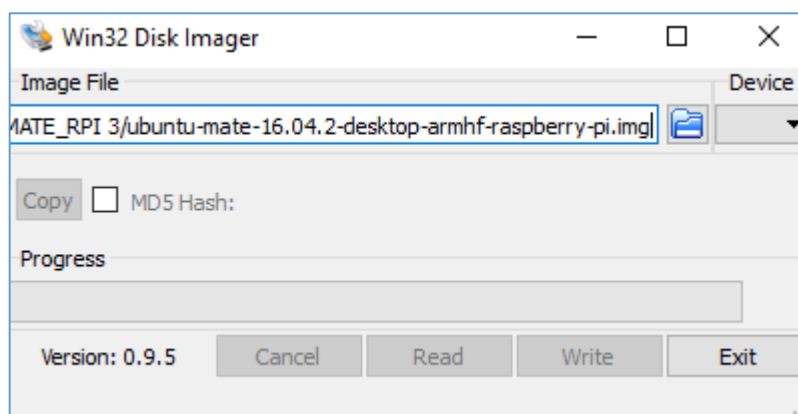


Figura 4.11: Cargando la imagen de Ubuntu Mate al micro SD.

*Elaborado por: Investigador*

#### b) Conexión de dispositivos a la Raspberry PI

Se insertó la micro SD boteada a la placa RPI 3, seguidamente se conectó la Pantalla de 10.1”, teclado y el mouse. Finalmente se conectó la alimentación de 5V - 2.5A y la Raspberry Pi inicia la instalación de Ubuntu Mate, como se muestra en la Figura 4.12.

```
Starting User Sessions...
Starting /etc/rc.local Compatibility...
Starting getty on tty2-tty6 if dbus and logind are not available...
Starting LSB: Start NTP daemon...
Starting System Logging Service...
Starting Permit User Sessions...
[ OK ] Started Permit User Sessions.
My IP address is 192.168.3.41
[ OK ] Started getty on tty2-tty6 if dbus and logind are not available.
[ OK ] Started /etc/rc.local Compatibility.
Starting Getty on tty6...
[ OK ] Started Getty on tty6...
Starting Getty on tty5...
[ OK ] Started Getty on tty5...
Starting Getty on tty4...
[ OK ] Started Getty on tty4...
Starting Getty on tty3...
[ OK ] Started Getty on tty3...
Starting Getty on tty2...
[ OK ] Started Getty on tty2...
Starting Getty on tty1...
[ OK ] Started Getty on tty1...
Starting Serial Getty on ttyAMA0...
[ OK ] Started Serial Getty on ttyAMA0...
[ OK ] Reached target Login Prompts.
[ OK ] Started LSB: Start NTP daemon.
[ OK ] Started System Logging Service.
[ OK ] Reached target Multi-User System.
[ OK ] Reached target Graphical Interface.
Starting Update UTMP about System Runlevel Changes...
[ OK ] Started Update UTMP about System Runlevel Changes.
```

Figura 4.12: Inicio de la instalación de Ubuntu Mate en RPI 3

Elaborado por: Investigador

### c) Configuración de Ubuntu Mate

Después de la instalación de Ubuntu mate se procedió a la configuración general del sistema como: idioma, lugar, fecha, nombre del sistema y contraseña sudo. Si todos los pasos anteriores se realizaron correctamente entonces se podrá apreciar el escritorio de Ubuntu Mate, como se ilustra en la Figura 4.13.



Figura 4.13: Menú general de Ubuntu Mate

Elaborado por: Investigador

## 4.8.2 Eviacam para usar el módulo de la cámara RPI

Enable Viacam (eViacam) es un sustituto del mouse que mueve el puntero a partir del movimiento de la cabeza. Funciona en un ordenador (PC) y en las BSC con tan solo una webcam o módulo de cámara pequeña, pero sin elementos adicionales. Para poder usar esta aplicación en la tarjeta RPI 3 fue necesario instalar utilizando los siguientes comandos:

- `$ sudo add-apt-repository ppa:cesar-crea-si/eviacam`
- `$ sudo apt-get update`
- `$ sudo apt-get install eviacam`

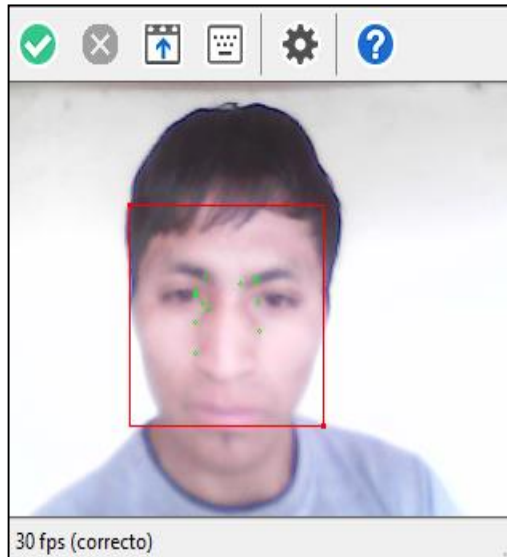
En la Figura 4.14 se muestra la instalación del software eviacam desde la consola de Ubuntu Mate.

```
hector@hector-desktop:~$ sudo apt-get install eviacam
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no
son necesarios.
 apt-clone archdetect-deb dpkg-repack gir1.2-json-1.0 gir1.2-timzone-1.0
 gir1.2-xkl-1.0 grub-common libdebian-installer4 libparted-fs-resize0
 libtimzone-data libtimzone1 os-prober python3-icu python3-pam rdate
 snap-confine
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
 libopencv-calib3d2.4v5 libopencv-features2d2.4v5 libopencv-flann2.4v5
 libopencv-highgui2.4v5 libopencv-legacy2.4v5 libopencv-ml2.4v5
 libopencv-objdetect2.4v5 libopencv-video2.4v5 opencv-data wx3.0-i18n
Se instalarán los siguientes paquetes NUEVOS:
 eviacam libopencv-calib3d2.4v5 libopencv-features2d2.4v5
 libopencv-flann2.4v5 libopencv-highgui2.4v5 libopencv-legacy2.4v5
 libopencv-ml2.4v5 libopencv-objdetect2.4v5 libopencv-video2.4v5 opencv-data
 wx3.0-i18n
0 actualizados, 11 nuevos se instalarán, 0 para eliminar y 3 no actualizados.
Se necesita descargar 8.369 kB de archivos.
Se utilizarán 33,3 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
```

Figura 4.14: Instalación de eviacam en Ubuntu Mate

Elaborado por: Investigador

Para confirmar la instalación de EVIACAM se dirigió al menú, aplicaciones, accesorios y se comprobó que la instalación tuvo éxito. Además, se verificó el funcionamiento de la aplicación conectando el módulo de la cámara al puerto CSI (Camera Serie Interface) de la RPI 3 y se ejecutó la aplicación instalada, como se puede observar en la Figura 4.15 la aplicación de EVIACAM detecta el rostro y con ello se pudo controlar el cursor del mouse.



*Figura 4.15: EVIACAM detecta el rostro*

*Elaborado por: Investigador*

### 4.8.3 Instalación de librerías para Python

El lenguaje de programación Python por defecto viene instalado en Ubuntu Mate en sus dos versiones Python 2.7 y Python 3.5. Para el desarrollo de nuestro sistema se usó Python 3.5 ya que brinda mayor cantidad de librerías que se encuentran disponibles en los repositorios de GIH-HUB de donde se puede descargar e instalar de acuerdo a las necesidades del usuario.

Para desarrollar nuestro sistema se utilizó diversas librerías las cuales se muestran en la Tabla 4.12, en la cual se describe brevemente cada una de ellas y se explica para que se empleó en el sistema. Más adelante se explicará detalladamente estas librerías y también se mostrarán los pasos necesarios para la instalación.

*Tabla 4.12: Librerías y programas empleadas en Python*

<b>Librerías Programas</b>	<b>Descripción</b>	<b>Aplicación en el Proyecto</b>
Tkinter	Es una librería de Python que dota de funciones para el desarrollo de interfaces de usuario GUI.	Se utilizó esta librería para desarrollar la ventana principal del teclado virtual con todas sus funciones.

eSpeak	Es un convertidor de texto a voz compacto de código abierto, está disponible para Linux y Windows en sus diferentes idiomas.	eSpeak se utilizó como sintetizador de voz, para escuchar mediante un altavoz el mensaje escrito por el paciente.
PyBluez	Es una librería que gestiona todos los recursos del sistema Bluetooth.	Se utilizó la librería de Pybluez para usar el Bluetooth de la tarjeta Raspberry Pi 3 y controlar el pequeño sistema domótico.
Autopy	Es un conjunto de herramientas GUI para Python, incluye diversas funciones que permiten controlar el teclado, cursor del mouse, entre otros.	Se utilizó para conocer la posición del cursor del mouse y desplazar por cada botón del teclado virtual con un tiempo determinado.
Smplayer	Es un reproductor multimedia gratuito para Windows o Linux, que tiene la facilidad de reproducir cualquier formato de audio o video.	Se utilizó esta aplicación para que el paciente pueda reproducir la música que desee.
Eviacam	Es una aplicación gratuita que sustituye al mouse y permite mover el cursor a partir del movimiento de la cabeza.	Se utilizó esta aplicación con la finalidad de que el paciente pueda mover el cursor del mouse con la cabeza, es un alternativa de comunicación para pacientes con etapas tardías de ELA.

*Elaborado por: Investigador*

A continuación, se explica cada una de las librerías empleadas en Python.

### **TKINTER**

Es una librería de interfaz gráfica de usuario (GUI) para el lenguaje de programación Python, se considera una librería estándar y viene instalado por defecto en Python. Existen otras alternativas disponibles para desarrollar un entorno gráfico visual, pero para nuestro caso se utilizó Tkinter de la versión 3.5 ya que cuenta con varios métodos



y funciones como: label, entry, button, toplevel, canvas, listbox, y otros que facilitaron el desarrollo del teclado virtual para el paciente.

Para utilizar esta librería en Python se debe importar con la siguiente línea de código:

```
from tkinter import *           // * importa todo el módulo de Tkinter
```

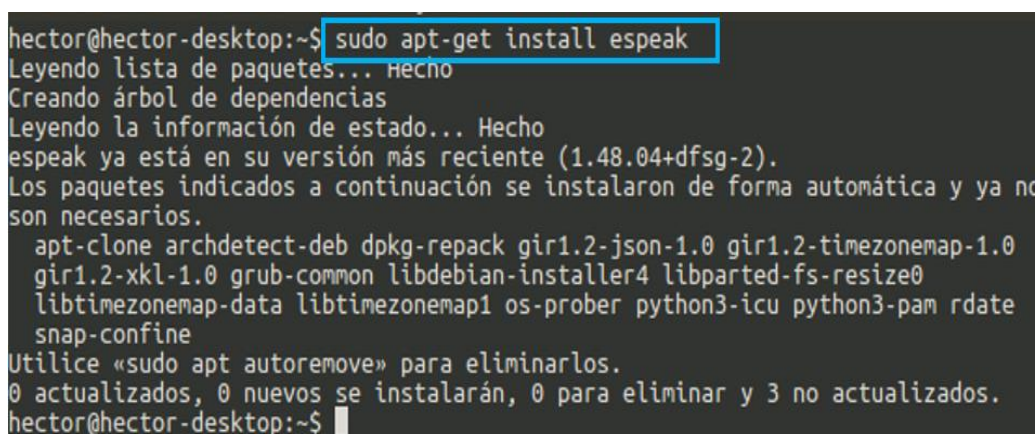
## ESPEAK

Para poder escuchar los mensajes escritos por el paciente, es necesario convertir el texto en audio, para ello se utilizó el sintetizador de voz compacto y de código abierto eSpeak, el cual posee características importantes y además brinda accesibilidad y facilidad para utilizar una vez que esté instalado adecuadamente, está disponible en diferentes idiomas como: inglés, español y otros.

Para utilizar eSpeak en Python fue necesario instalar la librería mediante el siguiente comando:

- `$ sudo apt-get install espeak` *// instala la librería del sintetizador de voz eSpeak*

En la figura 4.16 se muestra la instalación de la librería eSpeak desde la consola de Ubuntu Mate.



```
hector@hector-desktop:~$ sudo apt-get install espeak
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
espeak ya está en su versión más reciente (1.48.04+dfsg-2).
Los paquetes indicados a continuación se instalaron de forma automática y ya no
son necesarios.
 apt-clone archdetect-deb dpkg-repack gir1.2-json-1.0 gir1.2-timezonemap-1.0
 gir1.2-xkl-1.0 grub-common libdebian-installer4 libparted-fs-resize0
 libtimezonemap-data libtimezonemap1 os-prober python3-icu python3-pam rdate
 snap-confine
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 3 no actualizados.
hector@hector-desktop:~$
```

*Figura 4.16: Instalación de eSpeak*

*Elaborado por: Investigador*

En algunas versiones de Python la librería eSpeak viene instalado por defecto y se la puede utilizar mediante la importación de OS (`import os`), mientras que en otras versiones de Python es necesario instalar con el comando mencionado anteriormente (Figura 4.16). Para verificar su funcionamiento se realizó un ejemplo básico que permite

convertir texto en audio haciendo uso del sintetizador de voz eSpeak, como se muestra en la Figura 4.17.

```
from tkinter import *
import time
teclado = Tk()
a = StringVar()

def audio():
    voz = a.get()
    print(voz)
    import os
    os.system('espeak -ves+f4 "{0}" .format(voz))
    os.system('espeak -ves "{0}" .format(voz))

cuadro = Entry(teclado, textvariable = a).place (x=10, y=10)

boton = Button(teclado, text="AUDIO", command=audio).place(x=30, y=30)
teclado.mainloop()
```

Figura 4.17: Ejemplo del uso del sintetizador de voz (eSpeak)

Elaborado por: Investigador

En el ejemplo anterior (Figura 4.17), se creó una ventana con una entrada de texto y un botón que al momento de presionar llama al método creado con el nombre audio, la cual obtiene el texto escrito y convierte en voz gracias al sintetizador eSpeak. Para poder reproducir cualquier longitud de cadena de texto se debe usar el siguiente formato:

- *os.system('espeak -ves "{0}" .format(texto a reproducir)') // reproduce cualquier longitud de cadena gracias a {0} .format().*

También se puede elegir entre la voz de hombre (-ves) o mujer (-ves+f4) y los diferentes idiomas preferidas.

## PYBLUEZ

Es una librería que permite el acceso a los recursos del sistema Bluetooth mediante el lenguaje de programación Python. Esta librería se utilizó para poder acceder al Bluetooth de la tarjeta RPI 3 y vincular con el módulo Bluetooth que está conectado al microcontrolador externo para manipular el pequeño sistema domótico. PYBLUEZ se instaló utilizando los siguientes comandos:

- *\$ sudo apt-get install bluetooth libbluetooth-dev*
- *\$ sudo python3 -m pip install pybluez*

A continuación, en la Figura 4.18 se muestra la instalación de la librería PYBLUEZ desde la consola de Ubuntu Mate.

```
hector@hector-desktop:~/Escritorio$ sudo apt-get install bluetooth libbluetooth-dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
bluetooth ya está en su versión más reciente (5.41-0ubuntu3-xenial1.1).
Se instalarán los siguientes paquetes NUEVOS:
 libbluetooth-dev
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 55 no actualizados.
Se necesita descargar 163 kB de archivos.
Se utilizarán 477 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://ppa.launchpad.net/ubuntu-pi-flavour-makers/ppa/ubuntu xenial/main armhf libbluetooth-dev armhf 5.41-0ubuntu3-xenial1.1 [163 kB]
Descargados 163 kB en 1s (109 kB/s)
Seleccionando el paquete libbluetooth-dev previamente no seleccionado.
(Leyendo la base de datos ... 196089 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar ../libbluetooth-dev_5.41-0ubuntu3-xenial1.1_armhf.deb ...
Desempaquetando libbluetooth-dev (5.41-0ubuntu3-xenial1.1) ...
Configurando libbluetooth-dev (5.41-0ubuntu3-xenial1.1) ...
hector@hector-desktop:~/Escritorio$ sudo python3 -m pip install pybluez
The directory '/home/hector/.cache/pip/http' or its parent directory is not owned by the current user and the cache has been disabled. Please check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
The directory '/home/hector/.cache/pip' or its parent directory is not owned by the current user and caching wheels has been disabled. check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
Collecting pybluez
  Downloading PyBluez-0.22.zip (109kB)
    100% |#####| 112kB 246kB/s
Installing collected packages: pybluez
  Running setup.py install for pybluez ... done
Successfully installed pybluez-0.22
You are using pip version 8.1.1, however version 9.0.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
hector@hector-desktop:~/Escritorio$
```

Figura 4.18: Comandos usados para instalar PYBLUEZ

Elaborado por: Investigador

Para utilizar el Bluetooth de la tarjeta RPI 3 haciendo uso de la librería PYBLUEZ se realizó un ejemplo sencillo que se muestra en la Figura 4.19, donde se visualiza la importación, configuración y envío de datos.

```
import bluetooth
macBT = "20:16:05:09:04:65"
port = 1
BT= bluetooth.BluetoothSocket(bluetooth.RFCOMM)
BT.connect((macBT, port))

#data = ""
while 1:

    dato = input()
    if dato == 'a':
        BT.send('a')
    elif dato == 'b':
        BT.send('b')
    elif dato == 'c':
        BT.send('c')
    else:
        break

BT.close()
```

Figura 4.19: Ejemplo para usar el Bluetooth

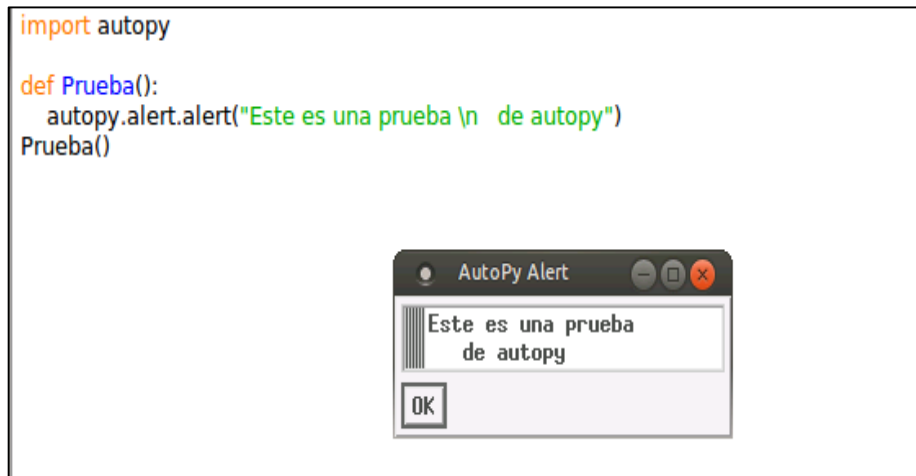
Elaborado por: Investigador

## AUTOPY

AutoPy es un conjunto de herramientas de automatización de GUI simple y multiplataforma para Python. Incluye funciones para controlar el teclado y el mouse, todo de manera multiplataforma, eficiente y simple. Se utilizó la librería de Autopy para controlar el desplazamiento del mouse por cada letra, simular el click izquierdo cada vez que el paciente desee escribir o seleccionar las diferentes funciones existentes en el teclado virtual. Para instalar la librería de Autopy se utilizó varios comandos debido a que es más complejo en comparación a las otras librerías instaladas anteriormente.

- Se inició instalando el GIT para compilar desde la última fuente en el repositorio de GITHUB:
  - *\$ sudo apt-get install git*
- Posteriormente se descargó la carpeta de Autopy con el comando:
  - *\$ git clone git://github.com/msanders/autopy.git*
- Además, se instaló todas las librerías auxiliares que Autopy requiere:
  - *\$ sudo apt-get install libx11-dev*
  - *\$ sudo apt-get install libxtst-dev*
  - *\$ sudo apt-get install libpng-dev*
- Después se accedió al directorio donde se encuentra la carpeta de Autopy descargada previamente:
  - *\$ cd autopy*
- Se compilo y construyo Autopy para la versión de Python 3.5 que se está utilizando:
  - *\$ python3 setup.py build*
- Finalmente se instaló Autopy con el comando:
  - *\$ sudo python3 setup.py install*

Después de finalizar la instalación se procedió a comprobar si la librería se ejecuta de forma correcta, para ello desde la consola de Python se escribió el comando *import autopy* obteniendo un resultado satisfactorio, seguidamente se verifico el funcionamiento adecuado realizando un ejemplo sencillo como se muestra en la Figura 4.20 y de la misma manera el resultado obtenido fue favorable.



*Figura 4.20: Ejemplo de prueba de Autopy*

*Elaborado por: Investigador*

#### **4.8.4 SMPlayer para reproducir música**

SMPlayer es un reproductor multimedia gratuito para Linux con códecs incorporados, que puede reproducir prácticamente cualquier formato de vídeo y audio. Para que el paciente con ELA pueda acceder a ver videos o escuchar música se instaló SMPlayer ya que brinda la facilidad de manipular sus iconos de acuerdo a las necesidades del usuario, logrando modificar el lugar donde se encuentra ubicado, cambiando el tamaño, entre otros.

La aplicación de SMPlayer se instaló desde la consola de Ubuntu Mate utilizando el siguiente comando:

- *\$ sudo apt-get install smplayer*

Desde Python se puede abrir cualquier aplicación o documento de texto usando librerías de acuerdo a las necesidades requeridas, es por ello que en la simulación del teclado virtual desarrollado para el paciente, se tiene la opción de escuchar música (Listen Music), cuando el paciente elija esta opción entonces se ejecutará la aplicación de SMPlayer cargado de canciones y videos favoritos, si desea cambiar de música, poner pausa o alguna otra actividad, el paciente podrá realizar sin ningún problema, ya que el cursor del mouse siempre estará desplazándose por las opciones de: stop, adelantar, retroceder, pausar o cerrar la aplicación, todos estos iconos se los modificó pensando en el bienestar y la forma más simple para que el paciente pueda utilizar la aplicación.

## 4.9 Interfaz gráfica para el paciente

Después de instalar las librerías necesarias, se desarrolló el teclado virtual en Python-Tkinter, la cual ofrece diversas funcionalidades, desde borrar, retroceder, hasta acceder a sub-ventanas en donde se encuentran mensajes predefinidos, control del pequeño sistema domótico, acceso a escuchar música, entre otros. En la Figura 4.21 se muestra la ventana principal del teclado virtual, con las palabras más usadas en español y letras ordenadas de acuerdo a la frecuencia que más se repiten. Cada botón tiene un tamaño ideal para mejor visibilidad y las letras que se van escribiendo en la entrada de texto (entry) se puede apreciar sin ninguna dificultad.

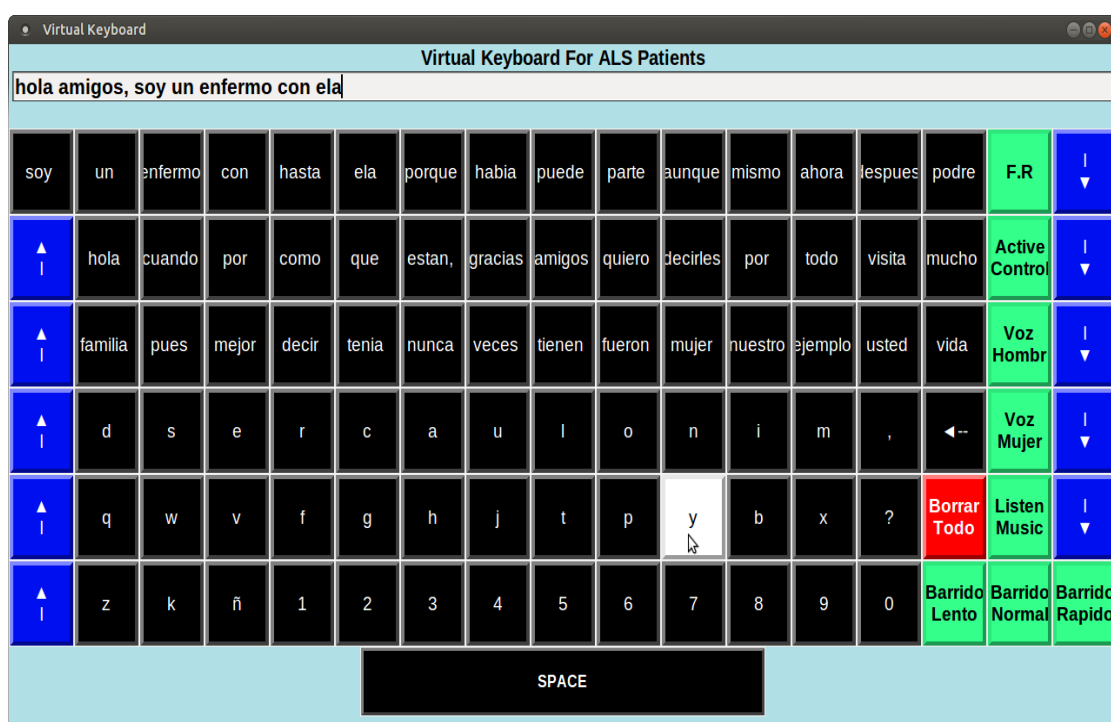


Figura 4.21: Teclado virtual para el paciente.

Elaborado por: Investigador

## 4.10 Implementación de la interfaz gráfica para el paciente.

La interfaz gráfica para el paciente con ELA se desarrolló de acuerdo a la siguiente estructura, en la cual se presentan todas las conexiones de los diferentes dispositivos empleados en el sistema, a continuación, en la Figura 4.22, se muestra el diagrama de conexión utilizado.

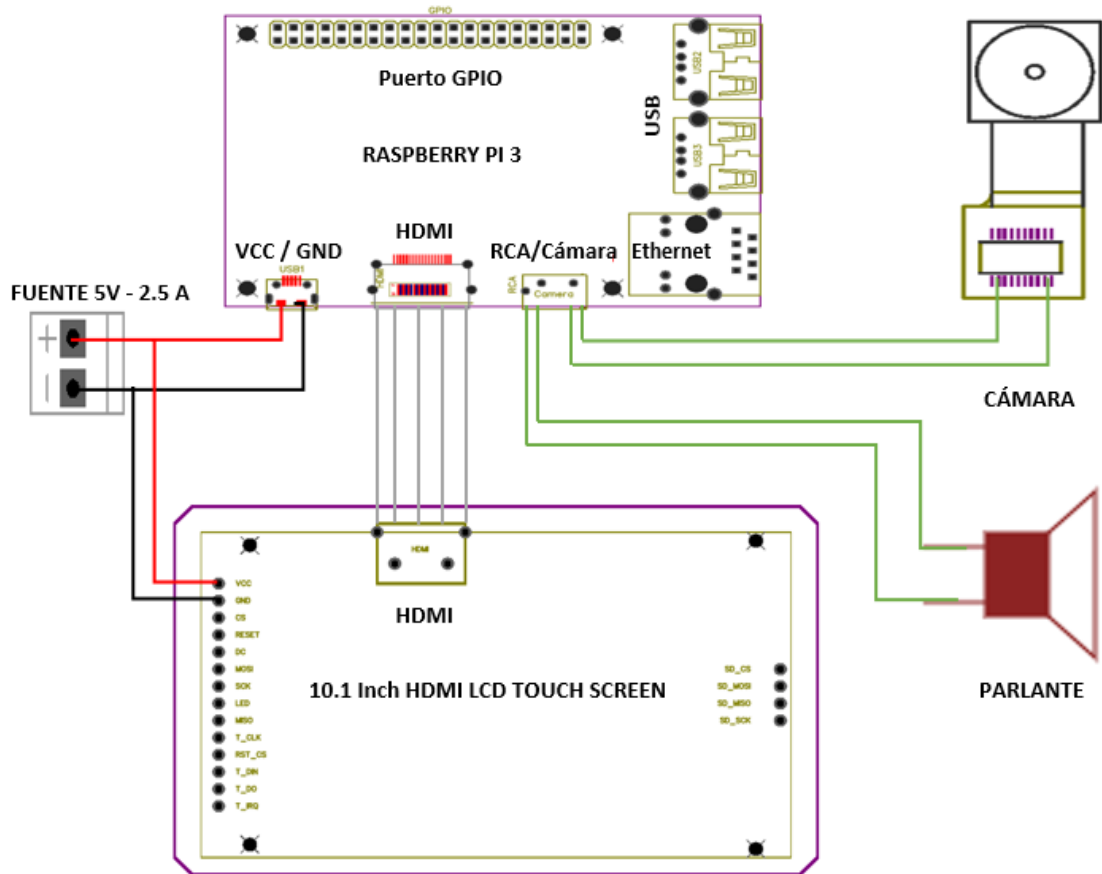


Figura 4.22: Diagrama de conexión de la interfaz para el paciente.

Elaborado por: Investigador

El proceso de desarrollo de la interfaz gráfica que se presenta en la Figura 4.22, consta de un controlador central que es la tarjeta Raspberry Pi 3, a ella se encuentran conectados módulos de dispositivos auxiliares, tales como: la pantalla Touch Screen de 10.1" a través de la salida HDMI, también se tiene conectado la cámara y los parlantes mediante el puerto CSI y el RCA respectivamente.

### Análisis de Consumo Energético

Para el funcionamiento correcto de la tarjeta RPI 3 es necesario una fuente de alimentación de 5 Vdc y una corriente mínima de 2 A, aunque lo ideal es utilizar una fuente con corriente de 2.5 A, de la misma manera la pantalla Touch Screen funciona al voltaje y corriente mencionado. El módulo de la cámara y el parlante trabajan utilizando la fuente de alimentación que proporciona la tarjeta RPI 3.

A continuación, se presenta en la Tabla 4.13 las características eléctricas de voltaje y corriente de cada dispositivo conectado a la tarjeta RPI 3, así como la potencia que



suministra cada uno de estos elementos y la potencia total que consume el circuito de la interfaz gráfica.

*Tabla 4.13: Potencia total que consume el circuito de la interfaz gráfica*

<b>Dispositivo</b>	<b>Voltaje</b>	<b>Corriente</b>	<b>Potencia</b>
Raspberry Pi 3	5 V	En reposo: 230 mA En rendimiento: 350 mA	1.2 W 1.8 W
Pantalla Touch Screen	5 V	2 A	10 W
Cámara RPI	5 V	250 mA	1.25 W
Parlantes	5 V	300 mA	1.5 W
Led indicador	5 V	15 mA	0.015 W
<b>Total</b>			14.5 W

*Elaborado por: Investigador*

La potencia total que consume el circuito de la interfaz gráfica es alta, además el sistema fue diseñado para que el paciente utilice en su casa, específicamente en la habitación donde se encuentre. Por tal razón, no es conveniente utilizar baterías como fuente de alimentación para el circuito, debido a que se puede conectar directamente a un punto de conexión eléctrica, similar a una TV o PC.

### **Conexiones de dispositivos a la RPI 3**

Como se ha mencionado anteriormente, la tarjeta RPI 3 es el núcleo principal de todo el sistema, a ella se encuentran conectados módulos externos complementarios que interactúan entre sí para el adecuado funcionamiento del mismo. En la Figura 4.23 se muestra la conexión de la cámara, los parlantes, la pantalla Touch Screen de 10.1” y el led indicador. La pantalla de 10.1” tiene agujeros en la parte posterior la cual permite adaptar de forma directa la tarjeta RPI 3, y evita que se produzca algún contacto indeseado o ruptura de cables al momento de mover la pantalla de un lugar a otro.





*Figura 4.23: Conexión de dispositivos a la placa RPI 3*

*Elaborado por: Investigador*

### **Prototipo final de la interfaz grafica**

Después de comprobar que todas las conexiones estén bien y verificar el correcto funcionamiento de la interfaz gráfica, se procedió a realizar el prototipo final utilizando el material de madera MDF, la cual tiene una dimensión de 30 cm de ancho, 28 cm de alto y una profundidad de 25 cm, en la parte frontal tiene una abertura que se ajusta a la pantalla Touch Screen de 10.1” (pulgadas), en la parte superior de la pantalla se puede observar la cámara de la RPI 3 que es muy parecida a la webcam de una computadora portátil, a la derecha de la cámara se encuentra un led indicador que se ha producido el movimiento del párpado o guiño del ojo. En la parte lateral derecha e izquierda se encuentra los parlantes que permiten escuchar los mensajes escritos por el paciente o la música deseada por el mismo. En la parte posterior se tiene un interruptor (switch con luz piloto) que permite encender o apagar el sistema, también está colocado un fusible como elemento de protección. Para finalizar en la parte interna del prototipo se encuentra todas las conexiones de la tarjeta RPI 3. En la Figura 2.24 se muestra el prototipo final de la interfaz gráfica para el paciente.



*Figura 4.24: Prototipo final de la interfaz gráfica*

*Elaborado por: Investigador*

#### **4.11 Implementación del sistema domótico**

La implementación del pequeño sistema domótico para el control de actividades por el paciente, se desarrolló de acuerdo a la siguiente estructura que se presenta en la Figura 4.25, donde se muestra todas las conexiones realizadas de los elementos que conforman el mismo.

El sistema principalmente cuenta con la placa microcontrolador NODEMCU como un núcleo central, a la cual se conecta el módulo Bluetooth HC-05 que recibe los datos proporcionados por la tarjeta RPI 3, como elementos de salida están conectados 2 servomotores para controlar la ventana y la puerta respectivamente, también se tiene el motor que cumple la función del ventilador, el led que simula el foco y un buzzer que actúa como alarma del paciente.

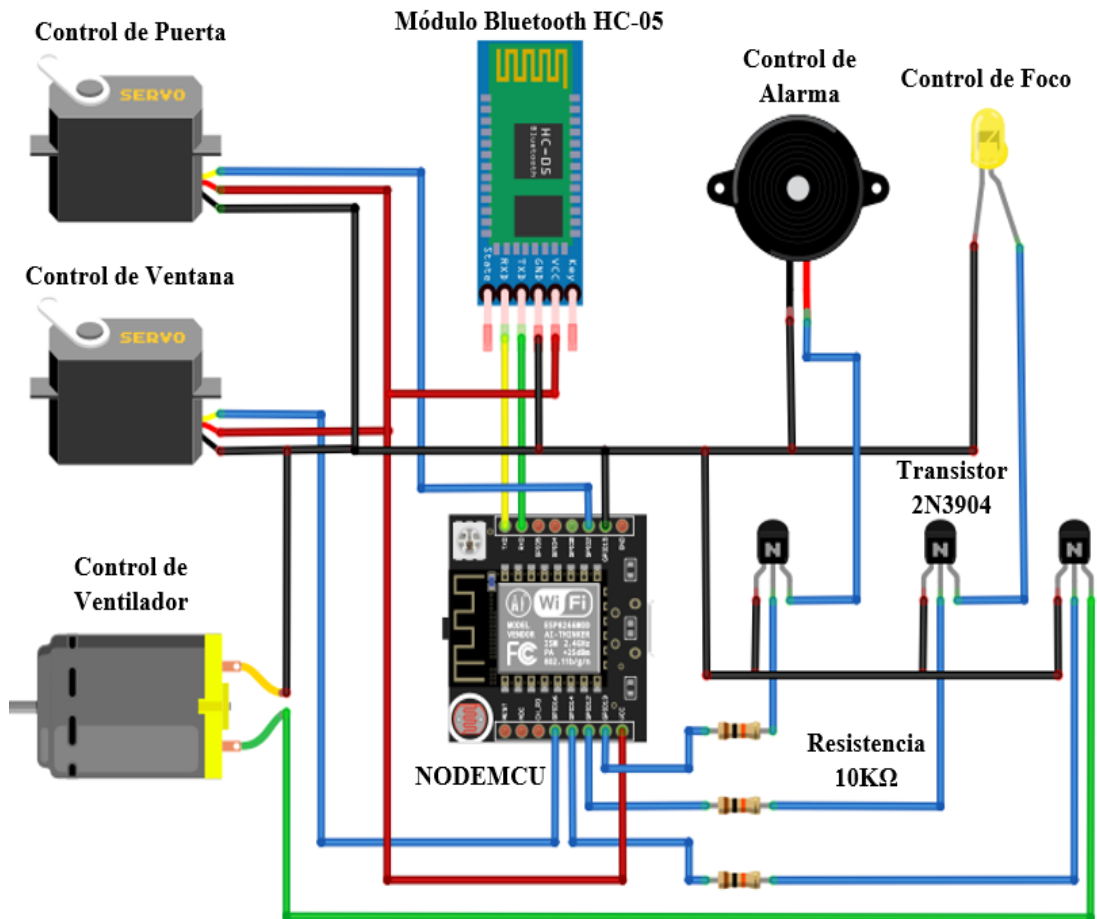


Figura 4.25: Esquema de conexión del sistema domótico

Elaborado por: Investigador

### Diagrama eléctrico del sistema domótico

Los pines GPIO del NODEMCU proporcionan un voltaje de salida máximo de 3.3 Vdc, por tal razón, es necesario utilizar una fuente externa con un voltaje superior a 5 Vdc. Con suficiente tensión los dispositivos que simulan la ventana, puerta, ventilador, led y alarma, funcionarían sin inconvenientes. El transistor trabaja como un interruptor entre, el pin del NODEMCU y la carga que se encuentra conectado a la fuente externa, la configuración que se realizó fue de Emisor Común. Para cargas inductivas (bobinas) se conectó un diodo en paralelo, polarizado inversamente, el diodo con esta conexión normalmente no hace nada. Pero cuando se deja de alimentar el circuito, la bobina dispara un pico de tensión inversa, esto dura solo unos microsegundos, pero es tiempo suficiente para dañar al transistor, el diodo hace que esa tensión fluya nuevamente a la bobina y no llegue al transistor. Para los servomotores no es necesario usar el transistor

debido a que estos motores tienen tarjetas controladoras que ayudan a trabajar con señales de PWM e incluso tienen su propio circuito de protección.

En la Figura 4.26 se muestra el diagrama eléctrico del sistema domótico con las conexiones de cada uno de los dispositivos.

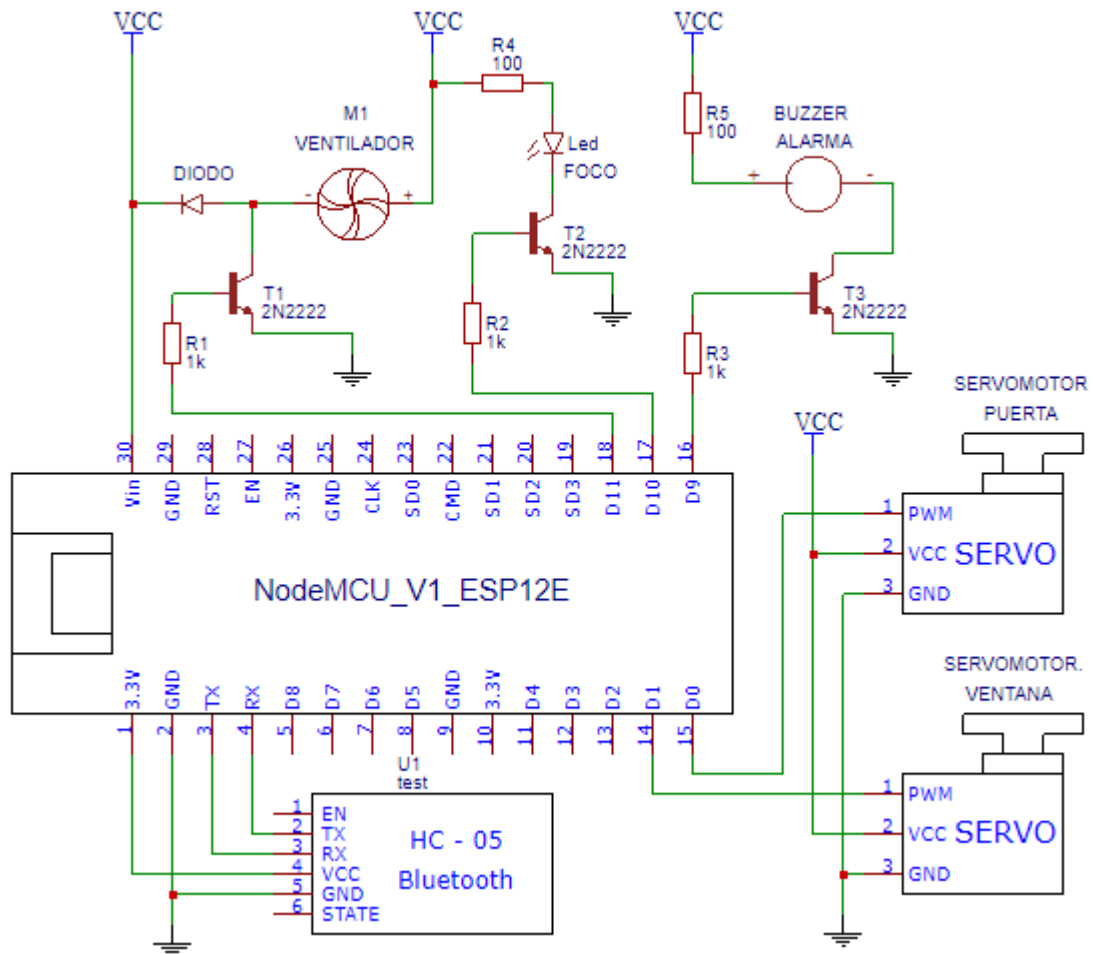


Figura 4.26: Diagrama eléctrico del sistema domótico

Elaborado por: Investigador

### Prototipo final del sistema domótico

Para el prototipo final del sistema domótico se realizó una pequeña maqueta que simula la habitación del paciente, como se muestra en la Figura 4.27. Se puede simular el abrir o cerrar la puerta, activar o desactivar la alarma del paciente, encender o apagar el ventilador, subir o bajar persianas (ventana) y finalmente encender, apagar o variar el estado del led. Esta manipulación se realiza de forma inalámbrica a través de la comunicación Bluetooth.





*Figura 4.27: Prototipo final del sistema domótico*

*Elaborado por: Investigador*

## **4.12 Pruebas de Funcionamiento**

Las pruebas de funcionamiento del sistema electrónico para facilitar la comunicación de los pacientes con ELA se realizaron en dos partes:

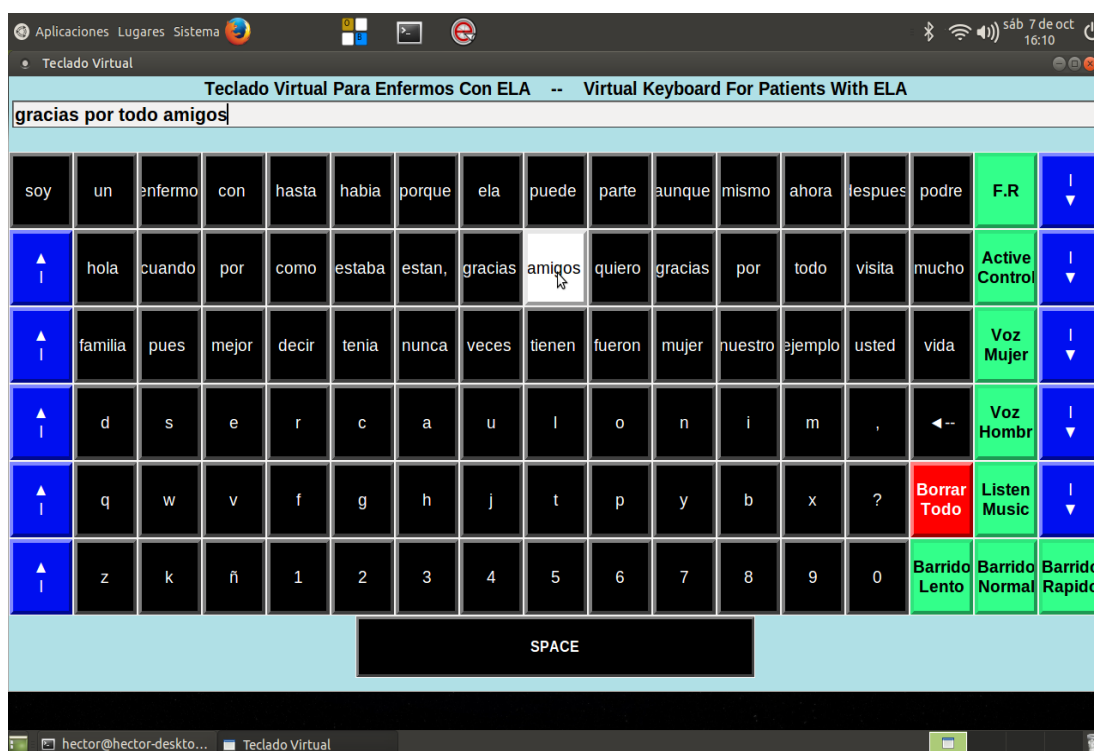
- Para pacientes con etapas finales de ELA, puesto que solo pueden mover sus párpados.
- Para pacientes con etapas intermedias de ELA, ya que pueden mover su cabeza.

### **4.12.1 Prueba realizada con el sensor TCRT5000**

#### **Teclado Virtual**

Esta prueba se enfoca en pacientes con etapas finales de ELA, para ello se debe ejecutar la aplicación para controlar el cursor del mouse (*Mouse.py*), esta aplicación permite desplazar el cursor del mouse de una letra a otra con un tiempo de 1 segundo, generando

un bucle infinito en la primera fila, si el paciente desea escribir la letra debe mover su párpado para que se produzca una interrupción y realice el click izquierdo, si la letra que desea escribir se encuentra en otra fila, entonces el paciente deberá mover su párpado cuando el cursor del mouse se encuentre en la flecha hacia abajo (↓), se debe repetir el mismo proceso para las demás filas, pero si desea subir fila deberá seleccionar la flecha hacia arriba (↑), de esta manera el paciente podrá acceder a todas las funciones del teclado virtual con solo el movimiento del párpado. La ventana principal del teclado virtual se muestra en la Figura 4.28.

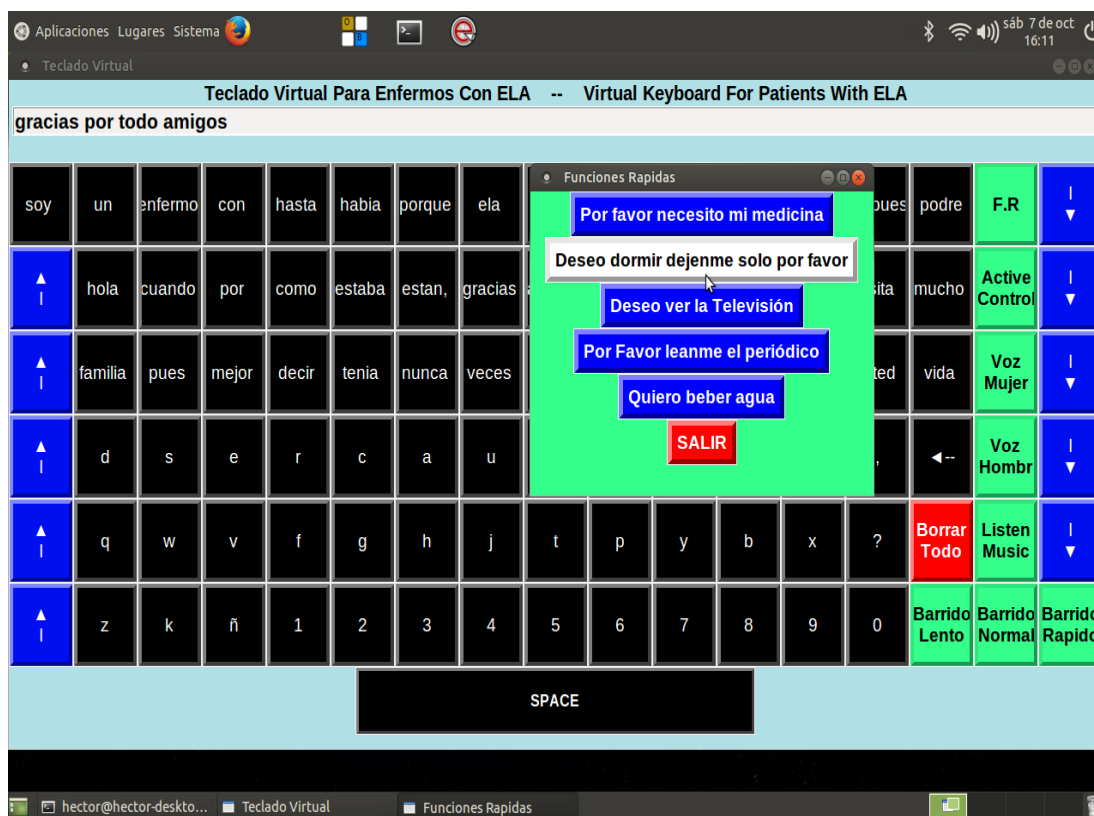


*Figura 4.28: Ventana principal del teclado virtual*

*Elaborado por: Investigador*

### **Funciones Rápidas (FR)**

La sub-ventana de funciones rápidas permite seleccionar y escuchar las necesidades básicas del paciente, el cursor del mouse se desplaza por cada mensaje formando un bucle infinito, si selecciona la opción de SALIR entonces la sub-ventana se cerrará y el cursor del mouse continuará en la primera fila. Esta prueba se realizó con 5 mensajes, pero se puede añadir más mensajes de acuerdo a las necesidades que requiera el paciente. En la Figura 4.29 se muestra la sub-ventana de funciones rápidas.



*Figura 4.29: Sub-ventana de funciones rápidas.*

*Elaborado por: Investigador*

## **Active Control**

Si el paciente selecciona la opción de active control, se abrirá la sub-ventana de control domótico, allí se tiene la opción para vincular el Bluetooth de la tarjeta RPI 3 con el módulo Bluetooth HC-05 que se encuentra conectado al microcontrolador externo (NODEMCU), de esta manera se podrá controlar las diferentes opciones, tales como, encender, apagar o variar la intensidad de la luz, también se puede activar o desactivar la alarma del paciente, encender o apagar el ventilador, abrir o cerrar la puerta y otras funciones existentes en ella. Al igual que en la opción de funciones rápidas (F.R) explicada previamente, el cursor del mouse se desplazará por cada opción de control domótico, formando un bucle infinito, y cuando el paciente desee cerrar esa ventana deberá seleccionar la opción de SALIR y el cursor del mouse regresará a la segunda fila. Del mismo modo, se puede añadir más opciones a la sub-ventana de control domótico según las necesidades requeridas por el paciente. En la Figura 4.30, se puede observar las opciones básicas para el control de dispositivos externos.

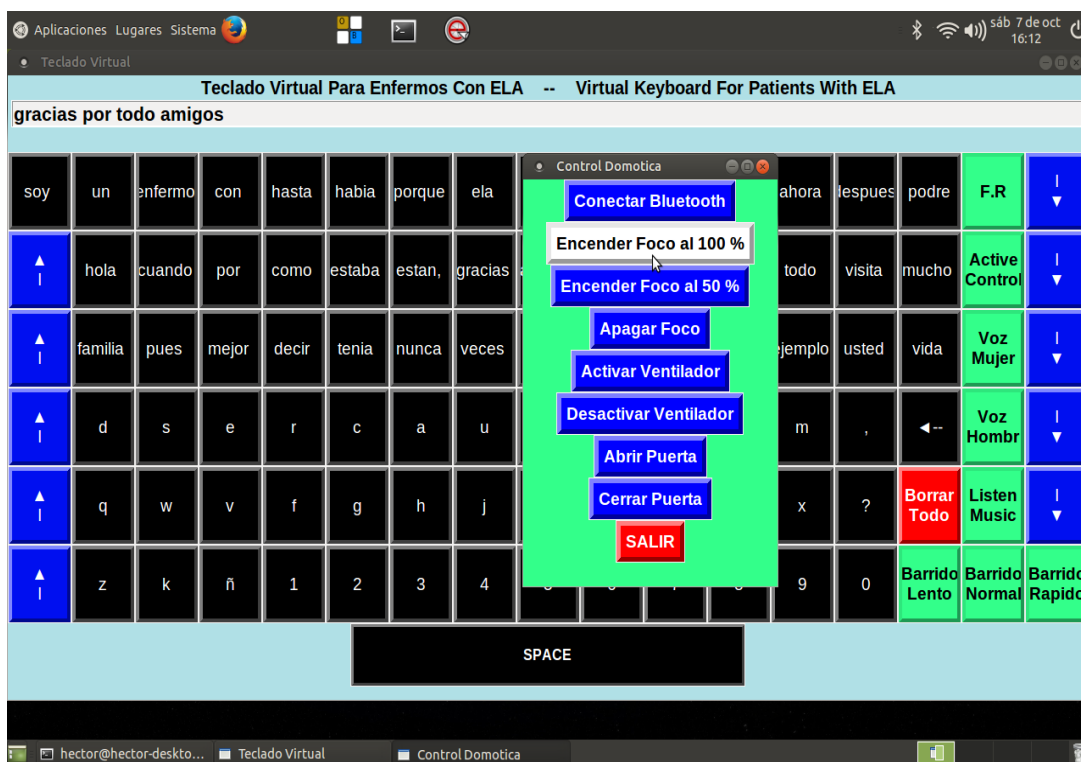


Figura 4.30: Sub-ventana de control domótico

Elaborado por: Investigador

## Escuchar Música

La aplicación de SMPlayer brinda la posibilidad de modificar sus iconos, cambiando de lugar o quitando algunos de ellos, por tal razón se tomó en cuenta los iconos más importantes y se ubicó en una sola fila para que el paciente tenga mayores facilidades de utilizar dicha aplicación.

Cuando el paciente seleccione la opción de escuchar música (*listen Music*), se ejecutará el reproductor SMPlayer cargado de canciones favoritos, el cursor del mouse genera un bucle infinito en los iconos de: reproducir, pausar, adelantar, retroceder y cerrar aplicación, de esta manera el paciente podrá controlar la aplicación de acuerdo a sus necesidades, si se elige la opción de cerrar aplicación entonces el reproductor se cerrará y el cursor del mouse retornara a la fila 5 del teclado virtual.

La prueba de funcionamiento de esta opción, resultado muy interesante e importante, ya que la mayor parte del tiempo el paciente se encuentra solo y gracias a esta opción podrá escuchar música para relajarse y disminuir la depresión. En la Figura 4.31, se muestra la aplicación de SMPlayer con la lista de canciones.



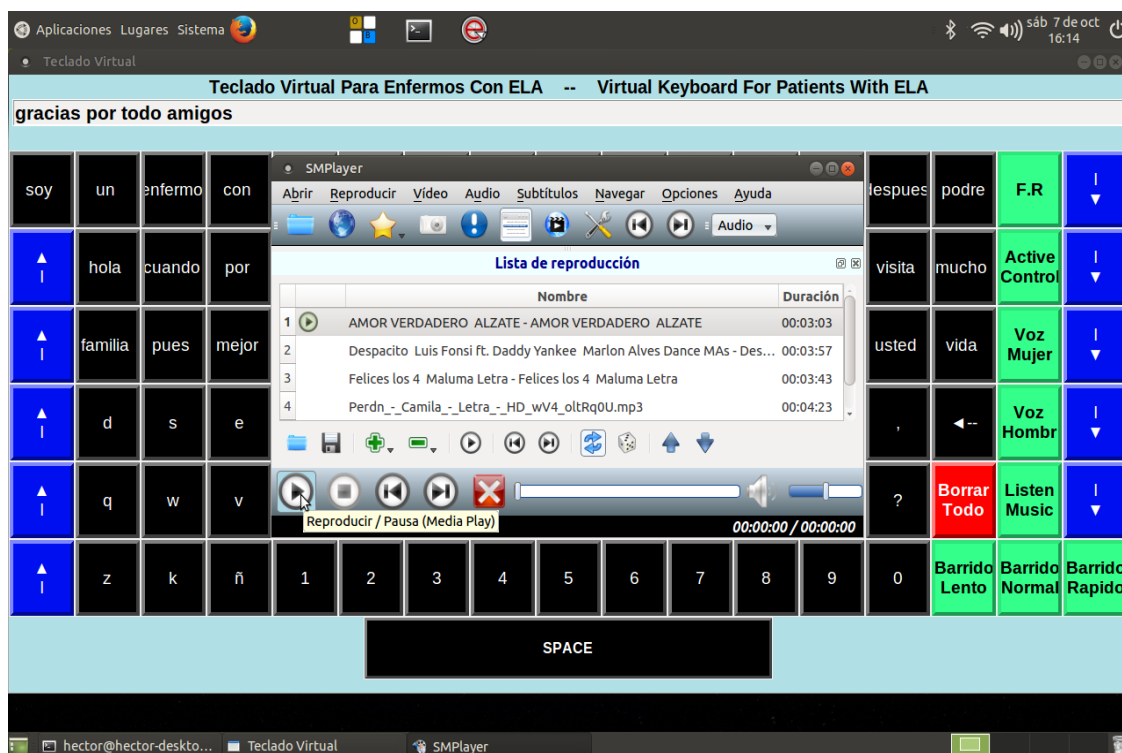


Figura 4.31: SMPlayer para escuchar música.

Elaborado por: Investigador

Continuando con las opciones del teclado virtual, también se puede elegir entre la voz masculina o femenina para escuchar los mensajes escritos por el paciente, además se tiene la opción que permite aumentar o disminuir el tiempo que tarda el cursor del mouse en desplazarse de una letra a otra, estas opciones están en:

- Barrido lento con 1500 ms y 2000 ms.
- Barrido normal con 1000 ms.
- Barrido rápido con 250 ms, 500 ms y 750 ms.

Estos tiempos son apropiados para que el paciente pueda adaptarse con mayor facilidad al sistema, dependiendo de su capacidad física podrá aumentar o disminuir la velocidad con que desea escribir y acceder a las diferentes funcionalidades del teclado virtual. De esta manera el paciente con ELA puede comunicarse con las personas de su alrededor y además realizar actividades cotidianas con solo el movimiento del párpado.

#### 4.12.2 Prueba realizada con la cámara RPI

Para pacientes con etapa tardía de ELA, no se debe ejecutar la aplicación de (*Mouse.py*) ya que el cursor del mouse es controlado mediante el movimiento de cabeza que detecta la aplicación de EVIACAM haciendo uso de la cámara de la RPI. En este caso el

proceso de escritura es diferente ya que el paciente podrá seleccionar la letra moviendo levemente su cabeza y escribir cuando el cursor del mouse se detenga fijamente por 1 segundo en la letra deseada.

De esta forma se puede acceder a las diferentes funciones del sistema, el proceso de comunicación y control de actividades se realizarán con mayor rapidez.

#### 4.13 Análisis de resultados

El sistema fue desarrollado usando Tkinter, la interfaz gráfica de Python. También se instaló varias librerías, tales como:

- Espeak, para convertir texto en audio (sintetizador de voz).
- PYBLUEZ, para poder usar el Bluetooth de la tarjeta RPI 3 y controlar dispositivos externos (Control Domótico).
- Autopy es un conjunto de herramientas GUI simple y multiplataforma para Python que incluye funciones para controlar el mouse.

Estas librerías fueron esenciales para el desarrollo y el correcto funcionamiento del sistema.

Para interpretar el funcionamiento del sistema se realizó pruebas con 10 personas, quienes no estaban afectados por ELA y usaron el sistema de forma voluntaria. Cada persona al terminar de utilizar el sistema, respondió la encuesta que se realizó en Formularios de Google, como se muestra en el Anexo E.

También es importante recordar que las pruebas se analizaron por etapas y además se verificó el funcionamiento tanto en el día como en la noche. Gracias a la ecuación (3) y los datos obtenidos de la encuesta, se calculó la efectividad del sistema en formato de porcentaje, como se muestra en la Tabla 4.14.

$$Efectividad_{sistema} = \frac{Resultado\ Real}{Resultado\ Esperado} * 100\ \% \quad (3)$$

#### Prueba realizada en el día.

El resultado real es la calificación mayor que se obtuvo de los encuestados, quienes tuvieron la apertura de calificar en el rango de 1 a 10 puntos y se tomó de dato el valor de 9 puntos, como se muestra en la Figura 4.32. El resultado esperado va ser siempre una constante de 10 puntos.

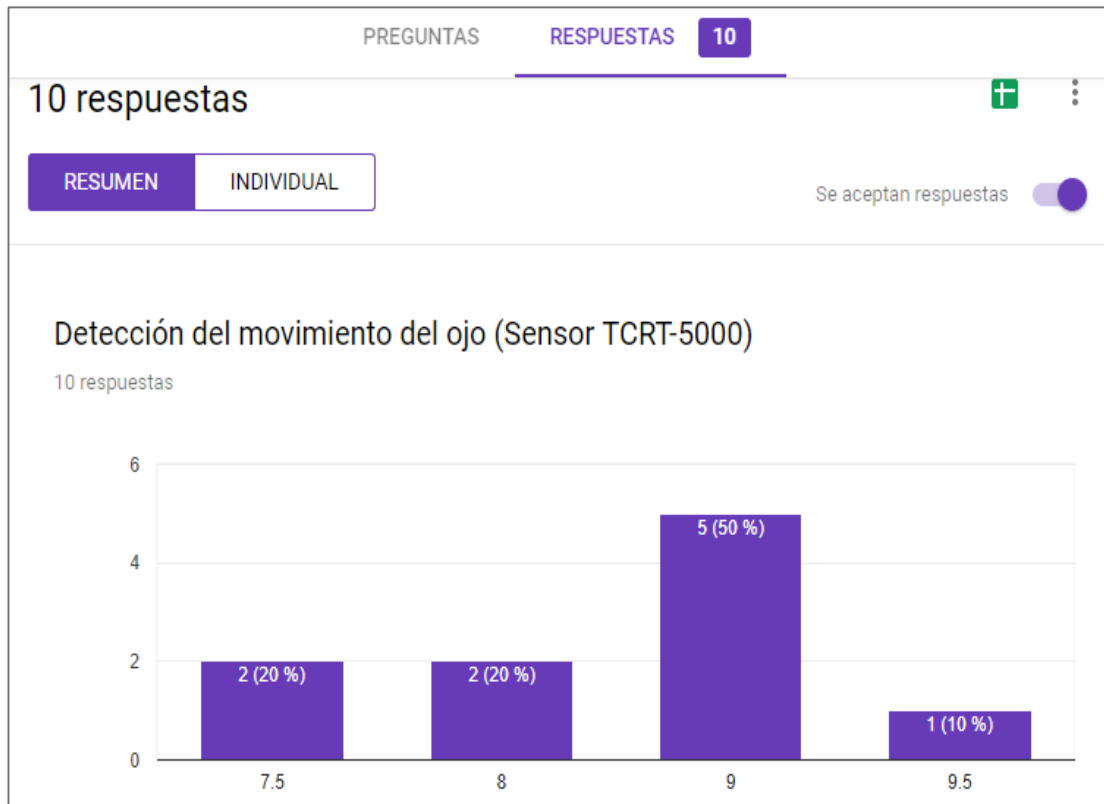


Figura 4.32: Respuesta mayor obtenida de los encuestados

Elaborado por: Investigador

Seguidamente se realiza el cálculo utilizando la ecuación (3), este proceso se repite para cada etapa que conforma el sistema.

$$\text{Sensor Proximidad}_{Día} = \frac{\text{Resultado Real}}{\text{Resultado Esperado}} * 100 \%$$

$$\text{Sensor Proximidad}_{Día} = \frac{9}{10} * 100 \%$$

$$\text{Sensor Proximidad}_{Día} = 0.9 * 100 \%$$

$$\text{Sensor Proximidad}_{Día} = 90 \%$$

#### Prueba realizada en la noche.

Para calcular la prueba realizada en la noche se realizó el procedimiento anterior.

$$\text{Sensor Proximidad}_{Noche} = \frac{\text{Resultado Real}}{\text{Resultado Esperado}} * 100 \%$$

$$\text{Sensor Proximidad}_{Noche} = \frac{8.5}{10} * 100 \%$$

$$\text{Sensor Proximidad}_{\text{Noche}} = 0.85 * 100 \%$$

$$\text{Sensor Proximidad}_{\text{Noche}} = 85 \%$$

Este proceso se cumplió para cada etapa que integra el sistema y el resultado se muestra a continuación en la Tabla 4.14.

*Tabla 4.14: Pruebas realizadas en el día y la noche.*

	<b>FUNCIÓN</b>	<b>Prueba realizada en el día</b>	<b>Prueba realizada en la noche</b>
Sensor de proximidad	Detectar movimiento del párpado	90%	85%
Cámara	Captar movimiento de cabeza	90%	85%
Acción del mouse	Desplazamiento lento	80%	80%
	Desplazamiento normal	80%	80%
	Desplazamiento rápido	80%	80%
	Click izquierdo	85%	85%
Interfaz gráfica	Simulación del teclado	90%	85%
	Sintetizador de voz	95%	90%
	Control domótico	80%	80%

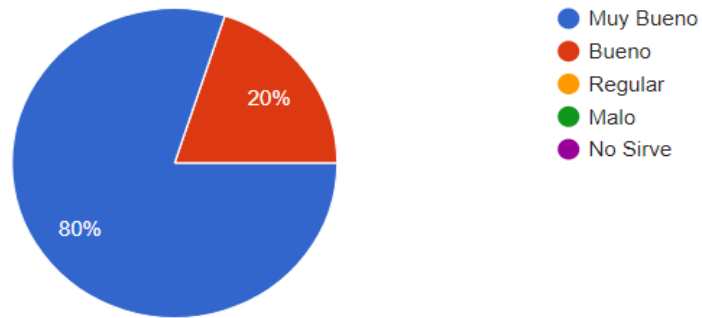
*Elaborado por: Investigador*

Las pruebas realizadas demuestran que la interfaz gráfica y la acción del mouse tanto en el día como en la noche funcionan correctamente, mientras que el uso de la cámara y el sensor de proximidad en la noche alteran de forma mínima el funcionamiento normal del sistema.

Finalmente se tomó el segundo dato de la encuesta, para analizar el resultado final del sistema de acuerdo al nivel de calidad, como se muestra en la Figura 4.33.

## ¿Como le pareció el funcionamiento general del Sistema?

10 respuestas



*Figura 4.33: Resultados obtenidos de personas que usaron el sistema*

*Elaborado por: Investigador*

Luego de realizar estas 10 pruebas se comprobó que el 80% (8 personas) calificaron como muy bueno, mientras que el 20% (2 personas) calificaron como bueno. Por esta razón el sistema tuvo mucha aceptación y fue destacado como un dispositivo seguro y confiable para ayudar a pacientes con esclerosis lateral amiotrófica (ELA).

### 4.14 Análisis económico del proyecto

#### 4.14.1 Costo de Hardware y Software

A continuación, en la Tabla 4.15 se detalla el costo de los dispositivos y componentes que fueron utilizados en el sistema electrónico para facilitar la comunicación y actividades cotidianas en pacientes con ELA.

*Tabla 4.15: Costo de Hardware y software*

Elemento	Cantidad	Valor Unitario	Valor Total
Modulo del Sensor TCRT500	1	\$ 8	\$ 8
Lentes	1	\$ 20	\$ 20
PIC 16F628A	1	\$ 6	\$ 6

Raspberry Pi 3	1	\$ 80	\$ 80
Cámara RPI	1	\$ 30	\$ 30
Micro SD 16Gb	1	\$ 15	\$ 15
Pantalla Touch Screen de 10.1"	1	\$ 106	\$ 106
Parlantes de PC	1	\$ 6	\$ 6
Caja de triplex	1	\$ 15	\$ 15
Modulo Bluetooth HC-05	2	\$ 8	\$ 16
Módulo ESP 8266	2	\$ 12	\$ 24
Batería Lipo	1	\$ 10	\$ 10
Diodos Leds	2	\$ 0.25	\$ 0.50
Servomotor	2	\$ 10	\$ 20
Buzzer	1	\$ 0.50	\$ 0.50
Ventilador	1	\$ 1	\$ 1
Cable jumper Hembra-Hembra	1	\$ 0.50	\$ 0.50
Caja de madera	1	\$ 10	\$ 10
Cable de internet de hilos	2 metros	\$ 1.50	\$ 1.50
Cargador de 5V a 2.5 A	2	\$ 10	\$ 20
Cable de Extensión	3 metros	\$ 3.50	\$ 3.50
Manguera Termorretráctil	2 metros	\$ 1.40	\$ 1.40
		<b>TOTAL</b>	<b>\$ 394.90</b>

*Elaborado por: Investigador*

#### 4.14.2 Costo inicial del diseño

Para la valoración del costo inicial del diseño se tomó en cuenta el total de horas trabajadas en el proyecto, el salario básico de un Ingeniero en Electrónica y Comunicaciones establecido por el Ministerio del Trabajo es: [45]

- Mensual: \$ 858
- Diario:  $\$ 415/21 = \$40.85$
- Hora:  $\$ 40.85/8 = \$ 5.10$

Se toma como tiempo estimado de trabajo un total de 110 horas de investigación, distribuidas en la realización del diseño, la programación y las pruebas de funcionamiento.

$$\text{Costo inicial del diseño} = \text{Total de horas trabajadas} * \text{valor de hora}$$

$$\text{Costo inicial del diseño} = 110 * \$ 5.10$$

$$\text{Costo inicial del diseño} = \$ 561$$

#### 4.14.3 Costo total del sistema

En la Tabla 4.16 se presenta el costo total del sistema electrónico para facilitar la comunicación y actividades cotidianas de pacientes con ELA.

*Tabla 4.16: Valor total para implementar el prototipo*

DESCRIPCIÓN	VALOR
Costo de hardware y software	\$ 394.90
Costo inicial del diseño	\$ 561
<b>Costo Total</b>	<b>\$ 955.90</b>

*Elaborado por: Investigador*

El valor inicial para implementar el prototipo del sistema electrónico para pacientes con ELA es de \$ 955.90 dólares, este valor es bajo en comparación a los dispositivos

comerciales que varían de 5 mil a 16 mil dólares. En la Tabla 4.17 se muestra una comparativa entre un dispositivo comercial de la empresa Tobii Dynavox y el sistema desarrollado.

*Tabla 4.17: Comparativa del dispositivo comercial y el sistema desarrollado*

	<b>Tobii I-15</b>	<b>Sistema desarrollado</b>
<b>Costo aproximado</b>	\$ 16.600	\$ 955.90
<b>Modo de Manipular</b>	✓ Movimiento Ocular	✓ Movimiento de párpados ✓ Movimiento facial
<b>Funciones</b>	<ul style="list-style-type: none"> <li>✓ Teclado virtual</li> <li>✓ Predictor de palabras</li> <li>✓ Sintetizador de voz</li> <li>✓ Escuchar música</li> </ul>	<ul style="list-style-type: none"> <li>✓ Teclado virtual</li> <li>✓ Sintetizador de voz</li> <li>✓ Funciones rápidas</li> <li>✓ Escuchar música</li> <li>✓ Sistema domótico</li> </ul>
<b>Pantalla</b>	Táctil	Táctil
<b>Disponibilidad</b>	Protegido	Libre (Open Source)

*Elaborado por: Investigador*

Como se mencionó anteriormente el costo del dispositivo comercial es demasiado alto en comparación al valor de nuestro sistema. Además, si desearíamos desarrollar más dispositivos el costo disminuiría, porque el costo inicial del diseño se descartaría. De esta manera el sistema realiza un aporte económico a la sociedad brindando servicios similares a los dispositivos comerciales, pero a precios bajos, permitiendo el acceso a las personas de bajos recursos económicos.



## CAPÍTULO 5

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1 Conclusiones

Al finalizar el presente proyecto de titulación se obtuvieron las siguientes conclusiones:

- El módulo del sensor TCRT5000 detecta movimientos del párpado o comisura lateral del ojo a una distancia de 5 mm y permite manipular el teclado de forma eficiente, mientras que el módulo de la cámara RPI captura el rostro facial para controlar el cursor del mouse y manejar el teclado con mayor rapidez.
- La interfaz gráfica para el paciente con Esclerosis Lateral Amiotrófica, es un sistema de comunicación alternativo y novedoso, el cual permite escribir mensajes que serán convertidos en audio mediante el sintetizador de voz, también ofrece la posibilidad de acceder a las diferentes funciones como mensajes predefinidos, escuchar música y control domótico, estas acciones se efectúan a partir del barrido por columnas que realiza el cursor del mouse con un tiempo  $t = 1s$ , este tiempo se puede aumentar o disminuir de acuerdo a la capacidad física del paciente.
- El sistema domótico suministra comodidad, seguridad y accesibilidad para personas con discapacidad motora seria, por consiguiente, los pacientes con Esclerosis Lateral Amiotrófica, tienen la posibilidad de utilizar esta herramienta para realizar actividades diarias tales como; manipular el estado de la luz, puerta, ventana y alarma a través de la comunicación Bluetooth, todas estas actividades se efectúan de manera eficiente y elevan la autoestima del paciente, generando

bienestar con ellos mismos y sobre todo no dependiendo en su totalidad de las personas que lo asisten.

- Los resultados obtenidos demostraron que el funcionamiento del sensor de proximidad y la cámara RPI en el día son más eficientes, mientras que las pruebas realizadas en la noche afectan en un 10% el funcionamiento normal del sistema, esto indica que se debe utilizar en lugares que tenga suficiente brillo para evitar inconvenientes.
- El prototipo desarrollado mediante software y hardware libre (Open Source), brinda la posibilidad de añadir a futuro más necesidades requeridas por el paciente, se cumplirá estos requerimientos sin modificar en su totalidad el código actual, con la finalidad de ayudar a personas de bajos recursos económicos, quienes por su situación monetaria tienen dificultad de acceder a estos dispositivos.

## 5.2 Recomendaciones

- La calibración del sensor TCRT5000 para que detecte el movimiento de la comisura lateral del ojo, se debe realizar antes de ejecutar la aplicación de *Mouse.py*, porque se establecería la red de comunicación Bluetooth y los datos que se emiten hacia la interfaz, podrían causar anomalías cuando se proceda hacer uso del teclado virtual.
- El sistema domótico presenta accesibilidad para manipular dispositivos externos, para lo cual es necesario vincular el Bluetooth de la tarjeta controladora central (RPI 3) con el módulo Bluetooth conectado a la placa NODEMCU antes de operar componentes domóticos, debido a que el sistema podría alterar y no ejecutar acciones solicitadas.
- La corriente que suministra el circuito es elevada, y la fuente de alimentación adecuada es de 5V a 2.5 A, tanto para la tarjeta Raspberry Pi 3, como para la pantalla Touch Screen, sino se tiene presente estas características eléctricas se

puede ver afectado el funcionamiento normal del sistema o destruir algún dispositivo externo conectado a la placa Raspberry Pi.

- En el micro SD se bootea la imagen ISO del sistema operativo, para ello se requiere que tenga una capacidad de almacenamiento superior a 16 GB y una velocidad de transmisión con núcleo 10, para de esta forma aprovechar todos los recursos de software con mayor rapidez y de manera eficiente, porque de lo contrario la rutina de la tarjeta RPI 3 se tornará lento y tedioso para el usuario.
- Las librerías de soporte instaladas para el lenguaje de programación Python deben ser verificadas mediante ejemplos básicos, puesto que en ocasiones se instala, pero en el instante de ejecutar el programa general se presenta errores y en ese caso es difícil identificar donde se originó.
- La programación mediante la creación y llamado de métodos, brinda facilidades lógicas al programador. Esta estrategia es importante porque se puede realizar diseños complejos con mayor seguridad y efectividad.
- Al manipular el sistema electrónico adecuadamente se tiene la certeza de prolongar la vida útil del mismo.

## BIBLIOGRAFÍA

- [1] National Institute of Neurological Disorders and Stroke, «Esclerosis Lateral amiotrófica,» Office of Communications and Public Liaison, 20 Diciembre 2016. [En línea]. Available: [https://espanol.ninds.nih.gov/trastornos/esclerosis\\_lateral\\_amiotrofica.htm](https://espanol.ninds.nih.gov/trastornos/esclerosis_lateral_amiotrofica.htm). [Último acceso: 28 Marzo 2017].
- [2] FUNDELA, «Fundación Española para el Fomento de la Investigación de la Esclerosis Lateral Amiotrófica.,» Información General, 2004 - 2014. [En línea]. Available: <http://www.fundela.es/ela/informacion-general/>. [Último acceso: 28 03 2017].
- [3] Redacción Médica, «Registro de Enfermos de ELA en el Ecuador,» SALUD PÚBLICA, 09 Marzo 2016. [En línea]. Available: <http://www.redaccionmedica.ec/secciones/salud-publica/50-casos-de-ela-registrados-en-el-ecuador-87334>. [Último acceso: 28 03 2017].
- [4] R. ÁLVAREZ, «El software que usa Stephen Hawking para comunicarse ya está disponible para todos,» 18 Agosto 2015. [En línea]. Available: <https://www.xataka.com/investigacion/el-software-que-usa-stephen-hawking-para-hablar-ya-esta-disponible-para-todos>. [Último acceso: 02 Abril 2017].
- [5] Tobii Dynavox, «Dispositivos y productos para enfermos de ELA,» Tobii, [En línea]. Available: <https://www.tobiidynavox.com/about/press-releases/2015/1/a-simple-and-fun-way-to-learn-eye-tracking-and-prepare-for-aac/>. [Último acceso: 02 04 2017].
- [6] LatamIsrael, «El dispositivo que puede cambiar la historia de los pacientes con ELA.,» Medicina, 18 08 2015. [En línea]. Available: <http://latamisrael.com/un-dispositivo-puede-cambiar-la-historia-de-los-pacientes-con-ela-en-todo-el-mundo/>. [Último acceso: 03 04 2017].

- [7] M. F. Vaca Muñoz, «Repositorio Institucional de la Universidad de las Fuerzas Armadas ESPE,» CINECA, 2016. [En línea]. Available: <http://repositorio.espe.edu.ec/handle/21000/12089>. [Último acceso: 03 04 2017].
- [8] M. G. Vintimilla Sarmiento, «Repositorio Institucional de la Universidad de las Fuerzas Armadas ESPE,» Junio 2014. [En línea]. Available: <http://repositorio.espe.edu.ec/handle/21000/9673>. [Último acceso: 05 04 2017].
- [9] D. P. Jarrín Zambrano, «Repositorio EPN - Mouse para personas con discapacidad motriz,» CINECA, 7 Agosto 2009. [En línea]. Available: <http://bibdigital.epn.edu.ec/handle/15000/1876>. [Último acceso: 05 04 2017].
- [10] L. Bossa Fernández , M. E. Abarca , E. S. Torres, G. Ramírez y M. d. J. Garcia, «Esclerosis Lateral Amiotrófica,» de *Manual para los pacientes con ELA*, pp. 10-15.
- [11] BlogSumed, «21 de junio día Mundial contra el ELA (Esclerosis lateral amiotrófica),» SUMED, 06 2015. [En línea]. Available: <http://blogsumed.es/blog/2015/06/15/21-de-junio-dia-mundial-contra-el-ela-esclerosis-lateral-amiotrofica/>. [Último acceso: 21 08 2017].
- [12] RedELA Investigación, «Que es ELA,» PLATAFORMA AFECTADOS DE ELA, [En línea]. Available: <http://www.plataformaafectadosela.org/acerca-de/>. [Último acceso: 12 05 2017].
- [13] «ELA (Esclerosis lateral amiotrófica),» Cuidate Plus, 21 09 2015. [En línea]. Available: <http://www.cuidateplus.com/enfermedades/neurologicas/ela.html#sintomas>. [Último acceso: 12 05 2017].
- [14] A. S. T. H. C. L. L. H. Y. Wang, «EMG BASED REHABILITATION SYSTEMS,» *IEEE Xplore Digital Library*, nº 15346850, pp. 1-4, 2015.
- [15] L. H. Otero, «Día Mundial Contra la ELA (Esclerosis Lateral Amiotrófica),» Cruz Roja Bizkaia, 21 06 2017. [En línea]. Available:

<http://salud.cruzrojabizkaia.org/2017/06/21/dia-mundial-contra-la-ela-esclerosis-lateral-amiotrofica/>. [Último acceso: 11 08 2017].

- [16] C. Basil, «Qué son los Sistemas Aumentativos y Alternativos de Comunicación (SAAC),» ARASAAC, [En línea]. Available: <http://www.arasaac.org/aac.php>. [Último acceso: 20 06 2017].
- [17] B. F. Rubin, «Tecnología para dar libertad a pacientes con parálisis,» CNET, 01 09 2014. [En línea]. Available: <https://www.cnet.com/es/noticias/tecnologia-para-dar-libertad-pacientes-con-paralisis-als/>. [Último acceso: 20 05 2017].
- [18] ELA, «Idean un sistema de comunicación para manejar el ordenador a partir de luces infrarrojas para pacientes con ELA,» Red Pacientes, 2013. [En línea]. Available: <http://redpacientes.com/social/posts/view/30438/9821>. [Último acceso: 03 06 2017].
- [19] R. Álvarez, «El software que usa Stephen Hawking para comunicarse ya está disponible para todos,» Xataka, 18 08 2015. [En línea]. Available: <https://www.xataka.com/investigacion/el-software-que-usa-stephen-hawking-para-hablar-ya-esta-disponible-para-todos>. [Último acceso: 10 07 2017].
- [20] AFP, «Stephen Hawking lanza el mayor programa de búsqueda de vida extraterrestre,» laprensalibre.cr, 20 07 2015. [En línea]. Available: <http://www.laprensalibre.cr/Noticias/detalle/33460/331/stephen-hawking-lanza-el-mayor-programa-de-busqueda-de-vida-extraterrestre>. [Último acceso: 02 08 2017].
- [21] P. Norloff, «Eye Tracking Technology is Improving Lives for ALS Patients,» LC TECHNOLOGIES, 01 06 2016. [En línea]. Available: <http://www.eyegaze.com/eye-tracking-technology-is-improving-lives-for-als-patients-2/>. [Último acceso: 06 08 2017].
- [22] W. Mason, «Oculus is working on eye tracking technology for the next generation of VR,» UPLOADVR, 08 10 2015. [En línea]. Available: <https://uploadvr.com/oculus-is-working-on-eye-tracking-technology-for-next-generation-of-vr/>. [Último acceso: 11 08 2017].

- [23] Tobii, «Power to be You,» TobiiDynavox, [En línea]. Available: <https://www.tobiidynavox.com/>. [Último acceso: 20 08 2017].
- [24] T. Tomilson, «Tobii Dynavox Augmentative and Alternative Communication devices now available,» DoSpace, 26 02 2016. [En línea]. Available: <https://www.dospace.org/tobii-dynavox-augmentative-and-alternative-communication-devices-now-available/>. [Último acceso: 25 08 2017].
- [25] W. Tomasi, «Sistemas de Comunicaciones Electrónicas,» Mexico, Person Education, 2003, p. 1.
- [26] Instituto de Educacion Secundaria Las Sabinas, Departamento de Tecnología, «Análisis de los sistemas electrónicos,» [En línea]. Available: <http://blog.educastur.es/tecnoaller/files/2011/02/apuntes-e-analogica.pdf>. [Último acceso: 02 09 2017].
- [27] M. F. Coca, A. García, L. Jordán, M. Rivero, S. Romero, K. Vanegas y G. Villaroel, «Sensor,» 2014. [En línea]. Available: <https://www.dui.uagrm.edu.bo/Informacion/Expociencia2014/1480.pdf>. [Último acceso: 10 09 2017].
- [28] CHERRY ELECTRICAL , «What is a Proximity Sensor,» Future Electronics, [En línea]. Available: <http://www.futureelectronics.com/en/sensors/proximity.aspx>. [Último acceso: 15 09 2017].
- [29] «Sensores de Proximidad,» DACS Electrónica, 2013. [En línea]. Available: <http://sensoresdeproximidad.galeon.com/>. [Último acceso: 28 09 2017].
- [30] «Sensor de infrarrojos,» Wiki de Robótica, 2015. [En línea]. Available: <http://wiki.robotica.webs.upv.es/wiki-de-robotica/sensores/sensores-proximidad/sensor-infrarrojos/>. [Último acceso: 01 10 2017].
- [31] J. E. Carrasco Freire, «Sistema electrónico de monitoreo inalámbrico del estado climático vial con señalética adaptativa para la reducción de velocidad en el paso

- lateral de Ambato, Universidad Técnica de Ambato,» 2016. [En línea]. Available: <http://repo.uta.edu.ec/handle/123456789/23067>. [Último acceso: 11 10 2017].
- [32] Universidad Politécnica de Valencia, «RASPERRY PI,» Blog Historia de la Informática, 18 12 2013. [En línea]. Available: <http://histinf.blogs.upv.es/2013/12/18/raspberry-pi/>. [Último acceso: 02 10 2017].
- [33] Øyvind Nydal Dahl, «What is BeagleBone,» BuildElectronicsCircuit, 25 06 2016. [En línea]. Available: <https://www.build-electronic-circuits.com/beaglebone/>. [Último acceso: 27 12 2017].
- [34] Rubén Velasco, «6 alternativas al Raspberry Pi para comprar este 2017,» RedesZone, 18 01 2017. [En línea]. Available: <https://www.redeszone.net/2017/01/18/6-alternativas-al-raspberry-pi-comprar-este-2017/>. [Último acceso: 17 10 2017].
- [35] R. Vega, «DOMÓTICA,» Ricveal, 28 02 2014. [En línea]. Available: <https://ricveal.com/blog/que-es-domotica/>. [Último acceso: 22 10 2017].
- [36] J. Marin, «Sistema Domotico y su funcionamiento.,» Blogger, 19 11 2012. [En línea]. Available: <http://julianamarin04.blogspot.com/2012/11/>. [Último acceso: 29 10 2017].
- [37] V. Boetzelaerstraat, «Domotica,» EGT Specialist in Domotica, [En línea]. Available: <http://www.e-g-t.nl/>. [Último acceso: 02 11 2017].
- [38] C. Velazquez, «Qué es la domótica,» EduTecno II, 04 06 2017. [En línea]. Available: <http://educaciontecnologicanes2017.blogspot.com/2017/07/que-es-la-domotica.html>. [Último acceso: 12 11 2017].
- [39] Free Software Foundation, «Qué es el software libre,» El sistema operativo GNU, [En línea]. Available: <https://www.gnu.org/philosophy/free-sw.es.html>. [Último acceso: 16 11 2017].
- [40] D. Janssen, «Wireless Communications,» Techopedia, [En línea]. Available: <https://www.techopedia.com/definition/10062/wireless-communications>. [Último acceso: 28 11 2017].



- [41] Anusha, «Wireless Communication: Introduction, Types and Applications,» Electronics Hub, 08 07 2017. [En línea]. Available: [https://www.electronicshub.org/wireless-communication-introduction-types-applications/#Types\\_of\\_Wireless\\_Communication\\_Systems](https://www.electronicshub.org/wireless-communication-introduction-types-applications/#Types_of_Wireless_Communication_Systems). [Último acceso: 03 12 2017].
- [42] P. Caffelli, «Qué es y cómo funciona el sistema Bluetooth,» etecnologia, [En línea]. Available: <https://etecnologia.com/gadgets/funcionamiento-bluetooth>. [Último acceso: 17 12 2017].
- [43] A. Cebrero, «TECNOLOGIAS INALAMBRICAS,» Wordpress, [En línea]. Available: <https://arianacebrero.wordpress.com/unidad-2/>. [Último acceso: 23 12 2017].
- [44] Home Automation PI IoT, «#Deal: Compatible 5MP camera for Raspberry Pi 3,» DYC Projects, [En línea]. Available: <https://diyprojects.io/deal-compatible-camera-5mp-raspberry-pi-3/#.WniERje23IV>. [Último acceso: 09 12 2017].
- [45] Ministerio de Trabajo, «Tabla salarial mínima,» 2018. [En línea]. Available: <https://drive.google.com/file/d/1SF5jJWfch5B9X3LqpmH2SDAm4HbPe22X/view>. [Último acceso: 27 01 2018].

# **ANEXOS**

## ANEXO A

### Software de Programación: PIC C Compiler

#### CÓDIGO DE PROGRAMACIÓN PARA LA ETAPA DE LENTES

*El software para la detección del movimiento de la comisura del ojo, está realizado bajo el lenguaje de programación PIC C Compiler. Se utiliza la interrupción externa para detectar dicho movimiento, también se usa la comunicación serial (RS232) para enviar el dato mediante el módulo Bluetooth HC-05, hacia la interfaz gráfica desarrollada en la tarjeta Raspberry PI 3.*

```
#include <16f628a.h>           // Incluimos la librería del PIC
#fuses XT                     // Fusible para el cristal Externo
#fuses NOPROTECT              // Código de lectura no protegido
#fuses NOWDT                  // No activado el perro guardián
#fuses INTRC                  // Oscilador RC (Importante)
#use delay(clock=4M)          // Reloj de 4 MHz
#use standard_io(b)           // Puerto B como salida Estándar
#use fast_io(a)               // Declaración rápida de puerto A

// Configuración de Comunicación Serial
#use rs232(baud=9600, xmit=pin_b2, rcv=pin_b1, parity=N, bits=8)

int cont;

// Configuración de la interrupción externa RB0
#INT_EXT
void envioDato()              // Método para enviar dato
{
    cont += 1;                 // Variable contador
    delay_ms(10);

    if(cont == 1)              // si contador es 1
    {
        putc('D');             // Envía dato (D)
        output_high(pin_b3);    // Enciende el led indicador
    }

    if(cont == 2)              // si contador es 2
    {
        putc('D');             // Envía Dato (D)
        output_low(pin_b3);     // Apaga led indicador
        delay_ms(250);         // retardo de 0,5 segundos
        cont = 0;              // resetea contador
    }
}
```

```

// Programa principal
void main()
{
    set_tris_a(0x00);           // Declaración de puerto A como Salida
    output_low(pin_a1);        // Pin A , en estado bajo
    enable_interrupts(GLOBAL); // Habilitamos la interrupción global
    enable_interrupts(INT_EXT); // Habilitamos la interrupción externa
    ext_int_edge(H_TO_L);      // Detecta flanco de bajada

    while (True)               // Bucle infinito
    {
        output_high(pin_a1);    // Realiza el encendido y apagado del Led
        delay_ms(500);          // es importante para que el programa
        output_low(pin_a1);     // funciones correctamente
        delay_ms(500);
    }
}

```

## ANEXO B

### Software de Programación: Python-Tkinter

#### CÓDIGO DE PROGRAMACIÓN PARA EL TECLADO VIRTUAL

*La simulación del teclado virtual esta realizado en Python-Tkinter, el mismo cuenta con varias librerías las cuales son configuradas y usadas mediante métodos para la correcta elaboración de la ventana principal y sub-ventanas. A continuación, se detallan las líneas de código que se emplearon.*

```
//Declaramos librerías usadas en Python
#! /usr/bin/env python
""" importamos las librerías usadas y con *
#importamos la librería completa de tkinter"""
from tkinter import *
import RPi.GPIO as GPIO
import bluetooth
import os

# CONFIGURACIÓN DE LOS SOCKETS PARA CONECTAR AL Bluetooth
CLIENTE
macBT = "20:16:05:09:04:65"
port = 1
sock = bluetooth.BluetoothSocket(bluetooth.RFCOMM)

def conectarBT():
    sock.connect((macBT, port))
    os.system('espeak -ves+f4 "bluetooth conectado")
    os.system("clear")

# CONFIGURACIÓN DE LOS PUERTOS GPIO DEL RPI
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(21, GPIO.OUT)

#CARGAMOS LA VENTANA A UNA VARIABLE Y AJUSTAMOS LA
DIMENSIÓN Y COLOR
teclado = Tk()
teclado.title("Teclado Virtual")
teclado ['bg'] = 'powder blue'
teclado.geometry("1280x660+0+0") # (ancho, alto, x, y)

# DECLARAMOS VARIABLES GLOBALES A USAR
sonido = StringVar()
dato = 0

# MÉTODO PARA LA VENTANA DE CONTROL DOMÓTICO
```

```

def controlDomotica():
    aux = Toplevel(teclado)
    aux.title("Control Domotica")
    aux.geometry("300x430+608+150")
    aux ['bg'] = '#33FF8A'

    controlD = [
        'Conectar Bluetooth',
        'Encender Foco al 100 %', 'Encender Foco al 50 %', 'Apagar Foco',
        'Activar Ventilador', 'Desactivar Ventilador', 'Abrir Puerta',
        'Cerrar Puerta', 'SALIR'
    ]

    for botonD in controlD:
        comando = lambda botonD_Tx = botonD: metodo_escribir(botonD_Tx)
        if botonD != "SALIR":
            Button(aux, text = botonD, command = comando,
                bg="blue" , fg="#ffffff", # color fondo
                font = ('arial', 14, 'bold'),
                activebackground="#ffffff",
                activeforeground="#000000",
                relief = "raised",
                padx=5, pady=5, bd=5).pack()

        if botonD == "SALIR":
            Button(aux, text = botonD, command = aux.destroy,
                bg="red" , fg="#ffffff", # color fondo
                font = ('arial', 14, 'bold'),
                activebackground="#ffffff",
                activeforeground="#000000",
                relief = "raised",
                padx=5, pady=5, bd=5).pack()

```

### # CREAMOS LA VENTANA PARA TECLAS DE FUNCIONES RÁPIDAS

```

def funcionesRapidas():
    aux2 = Toplevel(teclado)
    aux2.title("Funciones Rapidas")
    aux2 ['bg'] = '#33FF8A'
    aux2.geometry("400x300+608+150")

    funcionesR = [
        'Por favor necesito mi medicina', 'Deseo dormir dejenme solo por favor',
        'Deseo ver la Televisi3n', 'Por Favor leanme el peri3dico',
        'Quiero beber agua', 'SALIR'
    ]

    for teclaR in funcionesR:
        comando1 = lambda teclaR_Tx = teclaR: metodo_FR(teclaR_Tx)
        if teclaR != "SALIR":
            Button(aux2, text = teclaR, command = comando1,

```

```

bg="blue", fg="#ffffff", # color fondo
activebackground="#ffffff",
activeforeground="#000000",
relief = "raised", font = ('arial', 14, 'bold'),
padx=5, pady=5, bd=5). pack()

```

```

if teclaR== "SALIR":

```

```

    Button(aux2, text = teclaR, command = aux2.destroy,
           bg="red" , fg="#ffffff",
           activebackground="#ffffff",
           activeforeground="#000000",
           relief = "raised", font = ('arial', 14, 'bold'),
           padx=5, pady=5, bd=5).pack()

```

```

def metodo_FR(teclaR):

```

```

    if teclaR == "Por favor necesito mi medicina":
        os.system('espeak -ves "Por favor necesito mi medicina"')
        os.system("clear")

```

```

    if teclaR == "Deseo dormir dejenme solo por favor":
        os.system('espeak -ves "Deseo dormir dejenme solo por favor"')
        os.system("clear")

```

```

    if teclaR == "Deseo ver la TelevisiÃ³n":
        os.system('espeak -ves "Deseo ver la TelevisiÃ³n"')
        os.system("clear")

```

```

    if teclaR == "Por Favor leanme el periÃ³dico":
        os.system('espeak -ves "Por Favor leanme el periÃ³dico"')
        os.system("clear")

```

```

    if teclaR == "Quiero beber agua":
        os.system('espeak -ves "Quiero beber agua"')
        os.system("clear")

```

```

#CREAMOS EL METODO QUE RECIBE PARAMETROS DEL TECLADO
# Y EJECUTA ACCIONES

```

```

def metodo_escribir(tecla_Rx):

```

```

    if tecla_Rx == "F.R":
        funcionesRapidas()

```

```

    elif tecla_Rx == "â-2\nI" or tecla_Rx == "I\nâ-1/4" or tecla_Rx == "Barrido\nLento"
or tecla_Rx == "Barrido\nNormal" or tecla_Rx == "Barrido\nRapido":
        print ("")

```

```

    elif tecla_Rx == "Voz\nMujer":

```

```

    voz = sonido.get()
    os.system('espeak -ves+f4 "{0}" .format (voz))

elif tecla_Rx == "Voz\nHomb":
    voz = sonido.get()
    os.system('espeak -ves "{0}" .format (voz))

elif tecla_Rx == "â—,--":
    borrar = entry.index("end")
    entry.delete((borrar)-1)

elif tecla_Rx == "Borrar\nTodo":
    entry.delete(0, END)

elif tecla_Rx == "Listen\nMusic":
    os.popen("smplayer")
    os.system("clear")

elif tecla_Rx == "Active\nControl":
    controlDomotica()

elif tecla_Rx == "Conectar Bluetooth":
    conectarBT()

elif tecla_Rx == "Encender Foco al 100 %":
    sock.send('a')
    os.system('espeak -ves "Foco Encendido al 100 % "')
    os.system("clear")

elif tecla_Rx == "Encender Foco al 50 %":
    sock.send('b')
    os.system('espeak -ves "Foco Encendido al 50 % "')
    os.system("clear")

elif tecla_Rx == "Apagar Foco":
    sock.send('c')
    os.system('espeak -ves "Foco Apagado "')
    os.system("clear")

elif tecla_Rx == "Activar Ventilador":
    sock.send('d')
    os.system('espeak -ves "Ventilador Activado "')
    os.system("clear")

elif tecla_Rx == "Desactivar Ventilador":
    sock.send('e')
    os.system('espeak -ves "Ventilador Desactivado "')
    os.system("clear")

elif tecla_Rx == "Abrir Puerta":

```



```

sock.send('f')
os.system('espeak -ves "Puerta Abierta"')
os.system("clear")

elif tecla_Rx == "Cerrar Puerta":
    sock.send('g')
    os.system('espeak -ves "Puerta Cerrada"')
    os.system("clear")

elif tecla_Rx == "SPACE": # da un espacio
    entry.insert(END, ' ')

else:
    # Escribe la tecla pulsada
    entry.insert(END, tecla_Rx)

#DECLARAMOS LAS LETRAS PARA EL TECLADO EN UNA LISTA QUE VA
DESDE INDICE 0-37
letras = [
    'que ', 'muy ', 'ahora ', 'también ', 'hasta ', 'habia ', 'porque ', 'solo ', 'puede ', 'parte ',
    'aunque ', 'mismo ', 'ahora ', 'después ', 'por ', 'F.R', '\nâ-¼',
    'â-²\nI', 'hola ', 'estas ', 'para ', 'como ', 'estaba ', 'cuando ', 'gracias ', 'amigos ',
    'quiero ', 'donde ', 'espero ', 'sobre ', 'todos ', 'mucho ', 'Active\nControl', '\nâ-¼',
    'â-²\nI', 'familia ', 'pues ', 'mejor ', 'decir ', 'tenia ', 'nunca ', 'veces ', 'tienen ', 'fueron ',
    'mujer ', 'nuestro ', 'ejemplo ', 'usted ', 'por ', 'Voz\nMujer', '\nâ-¼',
    'â-²\nI', 'd', 's', 'e', 'r', 'c', 'a', 'u', 'l', 'o', 'n', 'i', 'm', ',', 'â-,,--', 'Voz\nHomb', '\nâ-¼',
    'â-²\nI', 'q', 'w', 'v', 'f', 'g', 'h', 'j', 't', 'p', 'y', 'b', 'x', '?', 'Borrar\nTodo', 'Listen\nMusic',
    '\nâ-¼',
    'â-²\nI', 'z', 'k', 'Ã±', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0', 'Barrido\nLento',
    'Barrido\nNormal', 'Barrido\nRapido',
    'SPACE'
]

# CREAMOS UNA ETIQUETA EN LA VENTANA CON SU TAMAÑO Y TEXTO
etiqueta = Label(teclado,
    text="Teclado Virtual Para Enfermos Con ELA",
    font = ('arial', 15, 'bold'), bg = 'powder blue', fg = 'black').grid(row = 0,
columnspan=90)

#CREAMOS UNA ENTRADA PARA TEXTO (ESCRIBIR MENSAJES)
entry = Entry(teclado, textvariable=sonido, width=105, font = ('arial', 16, 'bold'), fg =
'black' )
entry.grid(row=1, columnspan=20) #.grid permite ubicar y cargar el texto entry

var_Fila = 4
var_Columna = 0
#declaramos la variable fila con 3
#declaramos la variable columna con 0

# creamos nuestro for y pasamos la lista de letras [] a la variable tecla
# pasamos la variable tecla a otra variable(tecla_Tx) para enviar al metodo
#creado previamente, se usa el comando lambda por defecto

```

for tecla in letras:

```
command = lambda tecla_Tx=tecla: metodo_escribir(tecla_Tx)
```

```
if tecla != "SPACE":
```

```
    Button(teclado, text= tecla,  
           width=5, height=3,  
           font = ('arial', 14),  
           bg="#000000", fg="#ffffff", # bordes  
           activebackground="#ffffff",  
           activeforeground="#000000",  
           relief = "raised",  
           padx=4, pady=4, bd=4, # ancho de bordes  
           command = command).grid(row=var_Fila, column=var_Columna)
```

```
if tecla == "Borrar\nTodo":
```

```
    Button(teclado, text= tecla,  
           width=5, height=3,  
           font = ('arial', 14, 'bold'),  
           bg="red", fg="#ffffff",  
           activebackground="#ffffff",  
           activeforeground="#000000",  
           relief = "raised",  
           padx=4, pady=4, bd=4,  
           command = command).grid(row=var_Fila, column=var_Columna)
```

```
if tecla == "Voz\nHombr" or tecla == "Voz\nMujer" or tecla == "Listen\nMusic" or  
tecla == "Active\nControl" or tecla == "F.R":
```

```
    Button(teclado, text= tecla,  
           width=5, height=3,  
           font = ('arial', 14, 'bold'),  
           bg="#33FF8A", fg="#000000",  
           activebackground="#ffffff",  
           activeforeground="black",  
           relief = "raised",  
           padx=4, pady=4, bd=4,  
           command = command).grid(row=var_Fila, column=var_Columna)
```

```
if tecla == "Barrido\nLento" or tecla == "Barrido\nNormal" or tecla ==  
"Barrido\nRapido":
```

```
    Button(teclado, text= tecla,  
           width=5, height=3,  
           font = ('arial', 14, 'bold'),  
           bg="#33FF8A", fg="#000000",  
           activebackground="#ffffff",  
           activeforeground="black",  
           relief = "raised",  
           padx=4, pady=4, bd=4,  
           command = command).grid(row=var_Fila, column=var_Columna)
```

```

if tecla == "â-2\nI" or tecla == "I\nâ-1/4":
    Button(teclado, text= tecla,
           width=5, height=3,
           font = ('arial', 14),
           bg="#000ff0", fg="#ffffff",
           activebackground="#ffffff",
           activeforeground="#000000",
           relief = "raised",
           padx=4, pady=4, bd=4,
           command = command).grid(row=var_Fila, column=var_Columna)

```

```

if tecla == "SPACE":
    Button(teclado, text= tecla,
           width=50, height=3,
           font = ('arial', 12, 'bold'),
           bg="#000000", fg="#ffffff",
           activebackground="#ffffff",
           activeforeground="#000990",
           relief = "raised",
           padx=3, pady=3, bd=3,
           command = command).grid(row=10, columnspan=20)

```

# crea botones con numero de filas y columnas

```

var_Columna += 1
if var_Columna >16 and var_Fila ==4:
    var_Columna = 0
    var_Fila +=1

if var_Columna >16 and var_Fila ==5:
    var_Columna = 0
    var_Fila +=1

if var_Columna > 16 and var_Fila ==6:
    var_Columna =0
    var_Fila +=1

if var_Columna > 16 and var_Fila ==7:
    var_Columna =0
    var_Fila +=1

if var_Columna > 16 and var_Fila ==8:
    var_Columna =0
    var_Fila +=1

```

```

teclado.mainloop()           # ejecuta la ventana principal en un bucle infinito

```

## ANEXO C

### Software de Programación: Python-Tkinter

#### CÓDIGO DE PROGRAMACIÓN PARA CONTROLAR EL MOUSE

*Para controlar el desplazamiento del mouse por cada letra del teclado virtual se desarrolló un archivo ejecutable en Python. El mismo brinda la facilidad de conocer la posición exacta del mouse y moverlo hacia otro lugar mediante tiempos (delays), cada vez que se produzca una interrupción externa o dicho de otra manera el paciente envíe la señal mediante el movimiento del ojo, entonces se producirá el click izquierdo. A continuación, se detalla las líneas de código usado.*

```
#!/usr/bin/env python3
""" importamos las librerías a usar """
import RPi.GPIO as GPIO
import autopy as ap
from autopy.mouse import LEFT_BUTTON
import time

""" configuramos los pines GPIO de la RPI3 """
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(16, GPIO.OUT) # declaramos el GPIO 21 como salida
GPIO.setup(2, GPIO.IN) # declaramos el GPIO 2 como entrada para la interrupción

""" declaración de las variables a usar """
contador = 0
delay = 1
auxCont = 0
pos = 0
afila2 = 0
afila3 = 0
afila4 = 0
afila5 = 0
afila6 = 0
afila1R = 0
afila2R = 0
afila3R = 0
afila4R = 0
afila5R = 0

funcionesR = 0
salirFR = 0
amusica = 0
salirMusica = 0
domotica = 0
```

```
salirDomotica =0
```

```
""" Este método es usado para hacer click izquierdo y además obtiene el valor de la
posición actual del mouse, también asignamos valores a las variables y declaramos
como globales para que pueda cambiar su valor en el bucle infinito (While)
"""
```

```
def metodoClick(channel):
```

```
    global contador
```

```
    global afile2
```

```
    global afile3
```

```
    global afile4
```

```
    global afile5
```

```
    global afile6
```

```
    global afile1R
```

```
    global afile2R
```

```
    global afile3R
```

```
    global afile4R
```

```
    global afile5R
```

```
    global funcionesR
```

```
    global salirFR
```

```
    global amusica
```

```
    global salirMusica
```

```
    global domotica
```

```
    global salirDomotica
```

```
    global delay
```

```
    global auxCont
```

```
    pos = ap.mouse.get_pos() #optenemos la posicion del mouse
```

```
    print(pos)
```

```
    if pos == (1254,200): # ir a fila 2
```

```
        afile2 = 20
```

```
    if pos == (1178, 200): # ir a F.R
```

```
        funcionesR = 1
```

```
    if pos == (800,435): # Salir F.R
```

```
        salirFR = 2
```

```
    if pos == (38,280): # gresar a fila 1
```

```
        afile1R = 11
```

```
    if pos == (1254,280): # ir a fila 3
```

```
        afile3 = 30
```

```
    if pos == (1178,280): # ir a control domotica
```

```
        domotica = 3
```

```
    if pos == (760,570): # salir domotica
```

```
        salirDomotica = 4
```

```

if pos == (38,365): # gresar a fila 2
    afile2R = 22

if pos == (1254,365): # ir a fila 4
    afile4 = 40

if pos == (38,450): # gresar a fila 3
    afile3R = 33

if pos == (1254,450): # ir a fila 5
    afile5 = 50

if pos == (38,535): # gresar a fila 4
    afile4R = 44

if pos == (1178,535): # ir a musica
    amusica = 5

if pos == (540,530): # salir de musica
    salirMusica = 6

if pos == (1254,535): # ir a fila 6
    afile6 = 60

if pos == (38,620): # gresar a fila 5
    afile5R = 55

if pos == (1102,620): # barrido lento
    delay = 1.5

if pos == (1178,620): #barrido normal
    delay = 1

if pos == (1254,620): #barrido rápido
    auxCont +=1
    if auxCont == 1:
        delay = 0.75

    elif auxCont == 2:
        delay = 0.5

    elif auxCont == 3:
        delay = 0.25
        auxCont = 0

"""Pone el alto o bajo salida del GPIO 16 notificando que se produjo una interrupción
y al mismo tiempo realiza un click izquierdo del mouse"""
contador += 1
if contador == 1:

```

```

GPIO.output(16, True)
ap.mouse.click(LEFT_BUTTON)
elif contador == 2:
    GPIO.output(16, False)
    ap.mouse.click(LEFT_BUTTON)
    contador = 0

print (contador)

```

""" detecta una interrupción de 1 a 0 y llama al metodo click"""

```
GPIO.add_event_detect(2, GPIO.FALLING, callback = metodoClick, bouncetime = 200)
```

""" bucle infinito del barrido del mouse y declaramos las variables globales a 0 para verificar si se produjo otra interrupción """

```

while True:
    afile2=0
    salirFR=0
    funcionesR=0
    ap.mouse.move(38,200) #barrido de la fila 1
    time.sleep(delay)
    ap.mouse.move(114,200)
    time.sleep(delay)
    ap.mouse.move(190,200)
    time.sleep(delay)
    ap.mouse.move(266,200)
    time.sleep(delay)
    ap.mouse.move(342,200)
    time.sleep(delay)
    ap.mouse.move(418,200)
    time.sleep(delay)
    ap.mouse.move(494,200)
    time.sleep(delay)
    ap.mouse.move(570,200)
    time.sleep(delay)
    ap.mouse.move(646,200)
    time.sleep(delay)
    ap.mouse.move(722,200)
    time.sleep(delay)
    ap.mouse.move(798,200)
    time.sleep(delay)
    ap.mouse.move(874,200)
    time.sleep(delay)
    ap.mouse.move(950,200)
    time.sleep(delay)
    ap.mouse.move(1026,200)
    time.sleep(delay)
    ap.mouse.move(1102,200)

```

```
time.sleep(delay)
ap.mouse.move(1178,200)
time.sleep(delay)
ap.mouse.move(1254,200)
time.sleep(delay)
```

```
if funcionesR == 1:           #barrido de las funciones rápidas
    while True:
        ap.mouse.move(800,210)
        time.sleep(delay)
        ap.mouse.move(800,255)
        time.sleep(delay)
        ap.mouse.move(800,300)
        time.sleep(delay)
        ap.mouse.move(800,345)
        time.sleep(delay)
        ap.mouse.move(800,390)
        time.sleep(delay)
        ap.mouse.move(800,435)
        time.sleep(delay)
        if salirFR == 2:
            break
```

```
if afile2 == 20:             #barrido de la fila 2
    while True:
        afile3 = 0
        afile1R = 0
        domotica = 0
        salirDomnotica = 0
        ap.mouse.move(38,280) #botón para regresar a la fila 1
        time.sleep(delay)
        if afile1R == 11:
            break
        ap.mouse.move(114,280)
        time.sleep(delay)
        ap.mouse.move(190,280)
        time.sleep(delay)
        ap.mouse.move(266,280)
        time.sleep(delay)
        ap.mouse.move(342,280)
        time.sleep(delay)
        ap.mouse.move(418,280)
        time.sleep(delay)
        ap.mouse.move(494,280)
        time.sleep(delay)
        ap.mouse.move(570,280)
        time.sleep(delay)
        ap.mouse.move(646,280)
        time.sleep(delay)
```



```
ap.mouse.move(722,280)
time.sleep(delay)
ap.mouse.move(798,280)
time.sleep(delay)
ap.mouse.move(874,280)
time.sleep(delay)
ap.mouse.move(950,280)
time.sleep(delay)
ap.mouse.move(1026,280)
time.sleep(delay)
ap.mouse.move(1102,280)
time.sleep(delay)
ap.mouse.move(1178,280)
time.sleep(delay)
ap.mouse.move(1254,280)
time.sleep(delay)
```

```
if domotica == 3:      #barrido de control domótico
```

```
    while True:
```

```
        ap.mouse.move(760,210)
        time.sleep(delay)
        ap.mouse.move(760,255)
        time.sleep(delay)
        ap.mouse.move(760,300)
        time.sleep(delay)
        ap.mouse.move(760,345)
        time.sleep(delay)
        ap.mouse.move(760,390)
        time.sleep(delay)
        ap.mouse.move(760,435)
        time.sleep(delay)
        ap.mouse.move(760,480)
        time.sleep(delay)
        ap.mouse.move(760,525)
        time.sleep(delay)
        ap.mouse.move(760,570)
        time.sleep(delay)
```

```
    if salirDomotica == 4:
        break
```

```
if afila3 == 30:
```

```
    # barrido de la fila 3
```

```
    while True:
```

```
        afila4=0
```

```
        afila2R=0
```

```
        ap.mouse.move(38,365)
```

```
        #botón para regresar a la fila 2
```

```
        time.sleep(delay)
```

```
        if afila2R == 22:
```

```
            break
```

```
ap.mouse.move(114,365)
time.sleep(delay)
ap.mouse.move(190,365)
time.sleep(delay)
ap.mouse.move(266,365)
time.sleep(delay)
ap.mouse.move(342,365)
time.sleep(delay)
ap.mouse.move(418,365)
time.sleep(delay)
ap.mouse.move(494,365)
time.sleep(delay)
ap.mouse.move(570,365)
time.sleep(delay)
ap.mouse.move(646,365)
time.sleep(delay)
ap.mouse.move(722,365)
time.sleep(delay)
ap.mouse.move(798,365)
time.sleep(delay)
ap.mouse.move(874,365)
time.sleep(delay)
ap.mouse.move(950,365)
time.sleep(delay)
ap.mouse.move(1026,365)
time.sleep(delay)
ap.mouse.move(1102,365)
time.sleep(delay)
ap.mouse.move(1178,365)
time.sleep(delay)
ap.mouse.move(1254,365)
time.sleep(delay)
```

```
if afila4 == 40:           #barrido de la fila 4
    while True:
        afila3R=0
        afila5=0
        ap.mouse.move(38,450) #botón para regresar a la fila 3
        time.sleep(delay)
        if afila3R == 33:
            break
        ap.mouse.move(114,450)
        time.sleep(delay)
        ap.mouse.move(190,450)
        time.sleep(delay)
        ap.mouse.move(266,450)
        time.sleep(delay)
        ap.mouse.move(342,450)
        time.sleep(delay)
        ap.mouse.move(418,450)
```

```
time.sleep(delay)
ap.mouse.move(494,450)
time.sleep(delay)
ap.mouse.move(570,450)
time.sleep(delay)
ap.mouse.move(646,450)
time.sleep(delay)
ap.mouse.move(722,450)
time.sleep(delay)
ap.mouse.move(798,450)
time.sleep(delay)
ap.mouse.move(874,450)
time.sleep(delay)
ap.mouse.move(950,450)
time.sleep(delay)
ap.mouse.move(1026,450)
time.sleep(delay)
ap.mouse.move(1102,450)
time.sleep(delay)
ap.mouse.move(1178,450)
time.sleep(delay)
ap.mouse.move(1254,450)
time.sleep(delay)
ap.mouse.move(720,710) #space
time.sleep(delay)
```

```
if afila5 == 50:           #barrido de la fila 5
    while True:
        amusica=0
        salirMusica=0
        afila6=0
        afila4R=0
        ap.mouse.move(38,535) #botón para regresar a la fila 4
        time.sleep(delay)
        if afila4R == 44:
            break
        ap.mouse.move(114,535)
        time.sleep(delay)
        ap.mouse.move(190,535)
        time.sleep(delay)
        ap.mouse.move(266,535)
        time.sleep(delay)
        ap.mouse.move(342,535)
        time.sleep(delay)
        ap.mouse.move(418,535)
        time.sleep(delay)
        ap.mouse.move(494,535)
        time.sleep(delay)
        ap.mouse.move(570,535)
        time.sleep(delay)
```

```
ap.mouse.move(646,535)
time.sleep(delay)
ap.mouse.move(722,535)
time.sleep(delay)
ap.mouse.move(798,535)
time.sleep(delay)
ap.mouse.move(874,535)
time.sleep(delay)
ap.mouse.move(950,535)
time.sleep(delay)
ap.mouse.move(1026,535)
time.sleep(delay)
ap.mouse.move(1102,535)
time.sleep(delay)
ap.mouse.move(1178,535)
time.sleep(delay)
ap.mouse.move(1254,535)
time.sleep(delay)
ap.mouse.move(720,710) #space
time.sleep(delay)
```

```
if amusica == 5:           #barrido de control de música
```

```
    while True:
```

```
        ap.mouse.move(340,530)
        time.sleep(delay)
        ap.mouse.move(390,530)
        time.sleep(delay)
        ap.mouse.move(440,530)
        time.sleep(delay)
        ap.mouse.move(490,530)
        time.sleep(delay)
        ap.mouse.move(540,530)
        time.sleep(delay)
        ap.mouse.move(870,530)
        time.sleep(delay)
```

```
        if salirMusica == 6:
```

```
            break
```

```
if afile6 == 60:         #barrido de la fila 6
```

```
    while True:
```

```
        domotica=0
```

```
        salirDomotica=0
```

```
        afile5R=0
```

```
        ap.mouse.move(38,620) #botón para regresar a la fila 5
```

```
        time.sleep(delay)
```

```
        if afile5R == 55:
```

```
            break
```

```
        ap.mouse.move(114,620)
```

```
        time.sleep(delay)
```

ap.mouse.move(190,620)  
time.sleep(delay)  
ap.mouse.move(266,620)  
time.sleep(delay)  
ap.mouse.move(342,620)  
time.sleep(delay)  
ap.mouse.move(418,620)  
time.sleep(delay)  
ap.mouse.move(494,620)  
time.sleep(delay)  
ap.mouse.move(570,620)  
time.sleep(delay)  
ap.mouse.move(646,620)  
time.sleep(delay)  
ap.mouse.move(722,620)  
time.sleep(delay)  
ap.mouse.move(798,620)  
time.sleep(delay)  
ap.mouse.move(874,620)  
time.sleep(delay)  
ap.mouse.move(950,620)  
time.sleep(delay)  
ap.mouse.move(1026,620)  
time.sleep(delay)  
ap.mouse.move(1102,620)  
time.sleep(delay)  
ap.mouse.move(1178,620)  
time.sleep(delay)  
ap.mouse.move(1254,620)  
time.sleep(delay)  
ap.mouse.move(720,710) #space  
time.sleep(delay)

## ANEXO D

### Software de Programación: Arduino

#### CÓDIGO DE PROGRAMACIÓN PARA SISTEMA DOMÓTICO

*El siguiente código está desarrollado en el IDE de Arduino, para el control de dispositivos externos como: foco, ventilador, puerta, ventana y alarma mediante comunicación Bluetooth y usando el modulo microcontrolador NODEMCU.*

```
// Declaración de variables y librerías a usar

#include <Servo.h> // declaramos la librería del servomotor
int foco = 16; // declaramos variables a usar
int ventilador = 5; //GPIO 16 es D0 del Nodemcu (ver diagrama)
int alarma = 4;
Servo puerta; // declaramos variables usadas para el servo
Servo ventana;
int dato=0; // variable para recibir el dato

// configuración de pines
void setup() {

  puerta.attach(0);
  ventana.attach(2);
  Serial.begin(9600);
  pinMode(foco, OUTPUT);
  pinMode(ventilador, OUTPUT);
  pinMode(alarma, OUTPUT);
  puerta.write(40);
  ventana.write(80);

}

// bucle infinito ejecución del programa
void loop() {
  if(Serial.available()>0){ // si existe dato en el buffer
    dato=Serial.read(); // lee el dato y guarda e la variable

    switch (dato) { // si dato es:
      case 'a': // foco encendido al 100 %
        analogWrite(foco,1023);
        break;

      case 'b': // foco encendido al 50 %
        analogWrite(foco,512);
        break;
    }
  }
}
```

```

case 'c': // foco apagado
    analogWrite(foco,0);
    break;

case 'd': // ventilador activado
    digitalWrite(ventilador, HIGH);
    break;

case 'e': // ventilador desactivado
    digitalWrite(ventilador, LOW);
    break;

case 'f': // puerta abierta
    puerta.write(40);
    break;

case 'g': // puerta cerrada
    puerta.write(150);
    break;

case 'h': // persiana abierta
    ventana.write(50);
    delay(4000);
    ventana.write(80);
    break;

case 'i': // persiana cerrada
    ventana.write(120);
    delay(4000);
    ventana.write(80);
    break;

case 'j': // alarma activada
    digitalWrite(alarma, HIGH);
    break;

case 'k': // alarma desactivado
    digitalWrite(alarma, LOW);
    break;
}
}
}

```

## ANEXO E

### Encuesta

La siguiente encuesta presenta las preguntas que se realizaron a las personas que utilizaron el sistema.

#### Encuesta para verificar el funcionamiento del sistema

Califique en un rango de 1 a 10 puntos, el funcionamiento de cada etapa que integra el sistema

Detección del movimiento del ojo (Sensor TCRT-5000)

Tu respuesta \_\_\_\_\_

Captar movimiento de cabeza (Camará RPI)

Tu respuesta \_\_\_\_\_

Acción del mouse (Tiempo de barrido del mouse)

Tu respuesta \_\_\_\_\_

Acción del mouse (click izquierdo)

Tu respuesta \_\_\_\_\_



## Sintetizador de voz

Tu respuesta

---

## Control domótico

Tu respuesta

---

¿Como le pareció el funcionamiento general del Sistema?

- Muy Bueno
- Bueno
- Regular
- Malo
- No Sirve

**ENVIAR**

Nunca envíes contraseñas a través de Formularios de Google.

Este contenido no ha sido creado ni aprobado por Google. [Informar sobre abusos](#) - [Condiciones del servicio](#) - [Otros términos](#)

Google Formularios

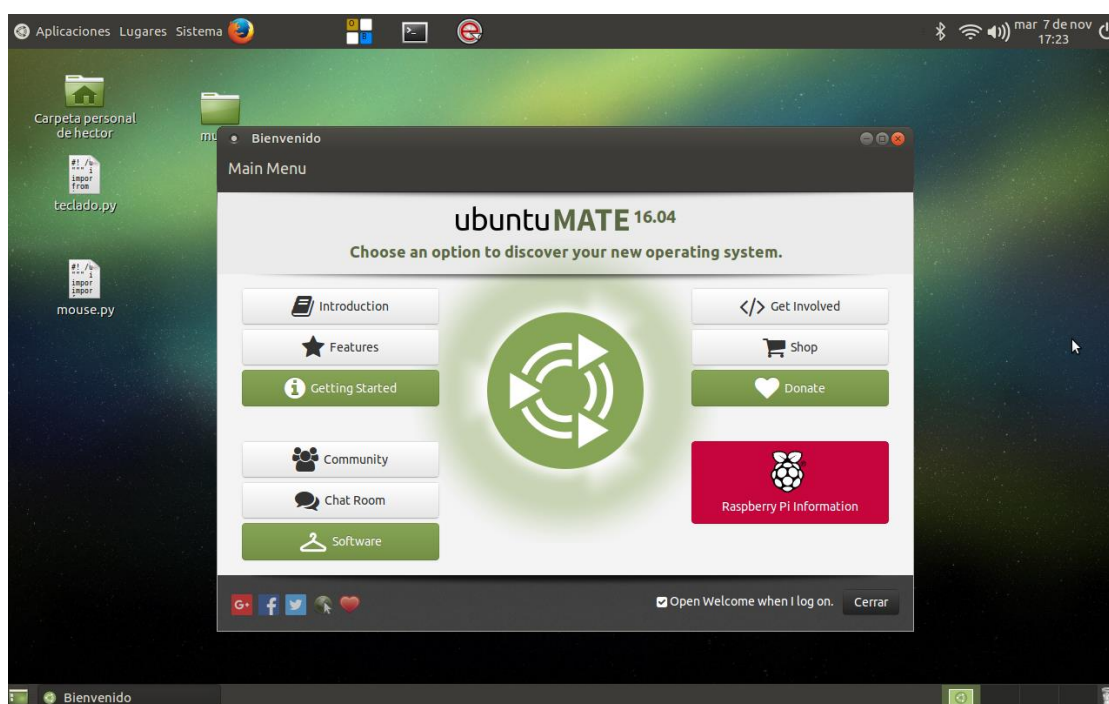
## ANEXO F

### Manual de usuario

El siguiente manual tiene como finalidad detallar todas las funcionalidades del teclado virtual y como utilizar de manera adecuada el sistema para pacientes con ELA.

#### 1. Encendido del sistema

Antes de encender el sistema se debe tomar en cuenta que todos los elementos se encuentren conectados correctamente, si es el caso se procederá a conectar la fuente de alimentación, entonces la pantalla se encenderá y presentará el escritorio de Ubuntu Mate de la siguiente manera, figura 1.



*Figura 1. Escritorio de Ubuntu Mate*

*Elaborado por: Investigador*

#### 2. Archivos ejecutables para usar el sistema

En el escritorio se encuentran 2 archivos ejecutables desarrollado en Python, el uno está con el nombre de **Teclado.py** y el otro con **Mouse.py**. El primero es la simulación del teclado virtual que al momento de ejecutar mostrara la ventana principal para el paciente con todas sus opciones, como se muestra en la figura 2. El segundo es el control del desplazamiento del mouse sobre el teclado.

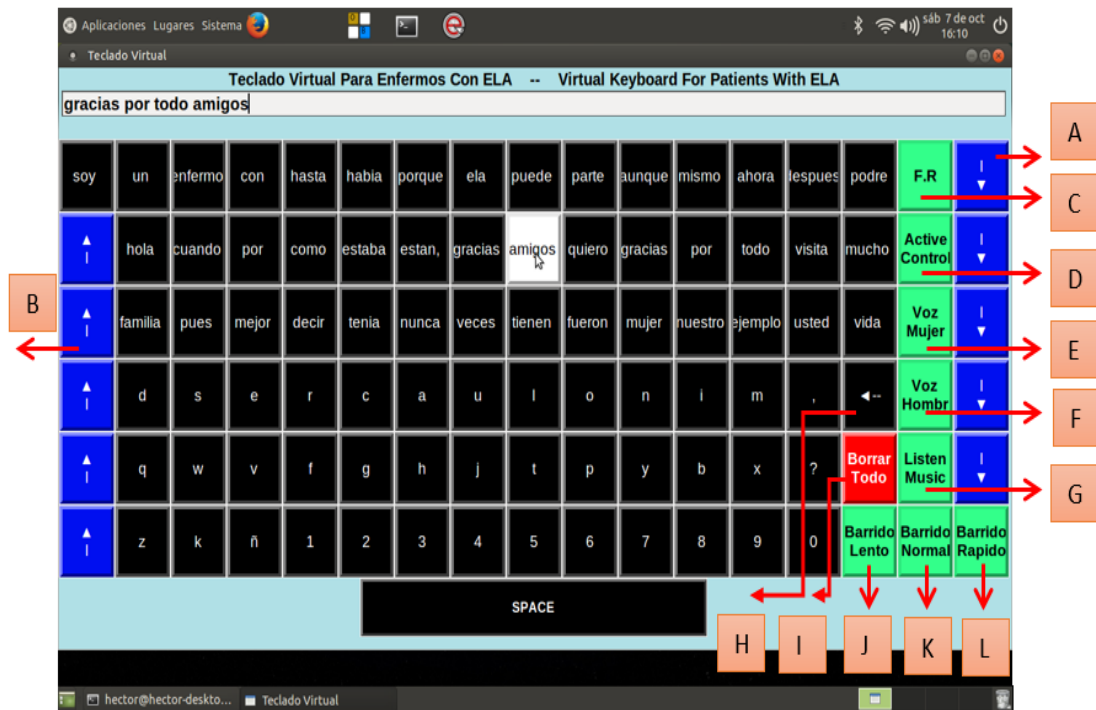


Figura 2. Simulación del teclado virtual y sus opciones

Elaborado por: Investigador

Las opciones que se encuentran en el teclado virtual son las siguientes:

- A. Tecla para bajar a la siguiente fila.
- B. Tecla para subir a la anterior fila.
- C. Tecla para acceder a la ventana de funciones rápidas (F.R) y escuchar mensajes predefinidas.
- D. Tecla de acceso a la ventana de control de dispositivos externos (foco, puerta, ventana, ventilador y alarma).
- E. Tecla del sintetizador de voz mujer.
- F. Tecla del sintetizador de voz hombre.
- G. Tecla de acceso a la ventana de músicas.
- H. Tecla para borrar letra.
- I. Tecla para borrar todo.
- J. Tecla para que el desplazamiento del mouse sea lento (1,5 – 2 ms).
- K. Tecla para que el desplazamiento del mouse sea lento (1ms).
- L. Tecla para que el desplazamiento del mouse sea rápido (0,75 – 0,5 – 0,25 ms).

### 3. Modos de usar el sistema

Existen 2 formas de usar el sistema:

- Con el movimiento del ojo (Sensor TCRT5000).
- Con movimiento de la cabeza (cámara RPI).

#### 3.1 Sensor TCRT5000

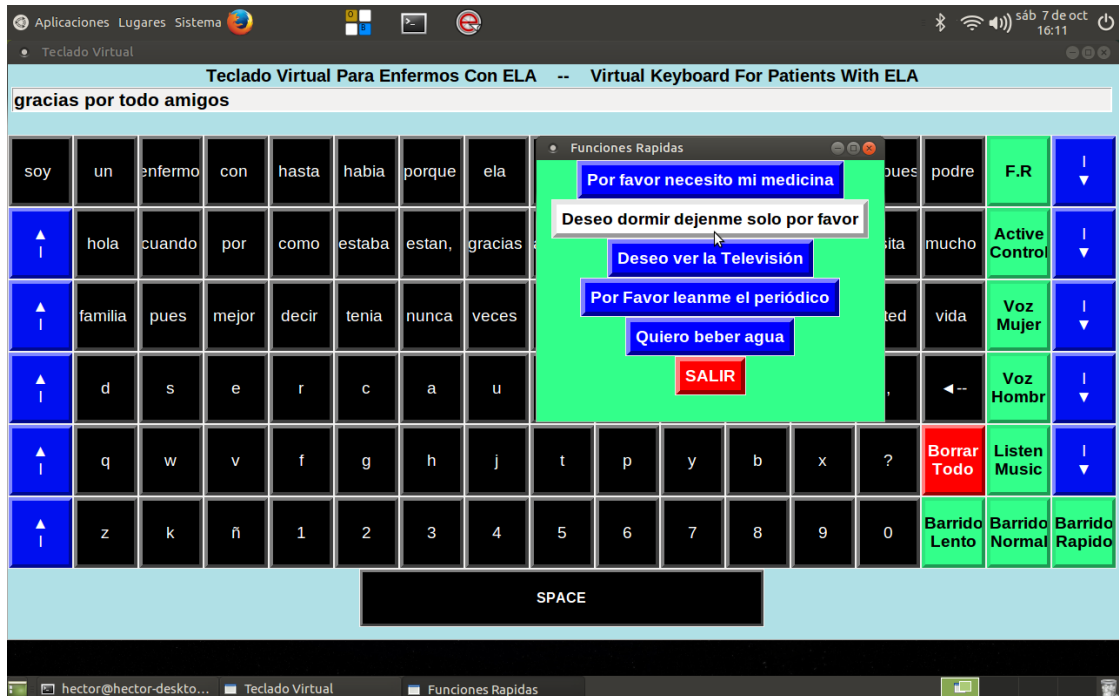
- ✓ Para utilizar el sistema con el movimiento del ojo, se debe ejecutar el archivo antes mencionado (*Mouse.py*), al momento de ejecutar dicho archivo, el cursor del mouse se desplazará por cada letra del teclado generando un bucle infinito en la primera fila.
- ✓ Posteriormente la persona que va utilizar el sistema, debe ponerse el prototipo de los lentes adecuadamente, ver figura 3.



*Figura 3. Prototipo de lentes ubicado adecuadamente.*

*Elaborado por: Investigador*

- ✓ Se debe encender el prototipo y verificar que los Bluetooths estén vinculados.
- ✓ Para verificar que el prototipo de los lentes está funcionando correctamente, al mover el ojo, el led indicador se encenderá y al mover otra vez el ojo el led indicador se apagará, este proceso del ON/OFF del led indicador debe repetirse cada vez que se mueva el ojo.
- ✓ El siguiente ejemplo muestra cómo se accedió satisfactoriamente a la ventana de funciones rápidas con el movimiento del ojo, ver figura 4.



*Figura 4. Acceso a la venta de funciones rápidas*

*Elaborado por: Investigador*

### 3.2 Cámara RPI

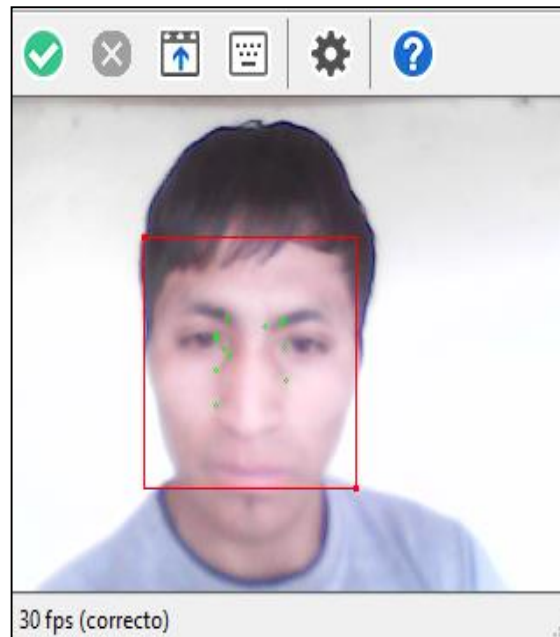
- ✓ Para usar el sistema con la cámara RPI, se debe ejecutar el software EVIACAM que se encuentra en la barra de herramientas del escritorio Ubuntu Mate, ver figura 4.



*Figura 4. Icono del software EVIACAM*


*Elaborado por: Investigador*

- ✓ Después se abrirá una ventana y detectará automáticamente el rostro de la persona, ver figura 5.



*Figura 5. Detección del rostro*

*Elaborado por: Investigador*

- ✓ Se puede acceder a la barra de herramientas para configurar múltiples opciones, entre ellas la velocidad del movimiento del mouse, el tiempo para que realice el click, entre otros.
- ✓ Después de la configuración debe iniciar eviacam, icono verde (  ) y tendrá el control absoluto del mouse, para manipular el teclado virtual y sus funciones explicadas anteriormente.

**Nota:** Si existiera errores como la calibración del sensor TCRT5000, la captura del rostro mediante EVIACAM, la ejecución del teclado virtual o algún problema de cualquier índole dentro del sistema de ayuda para pacientes con ELA. Deberá ponerse en contacto con el creador del mismo, el Sr: Collaguazo Chicaiza Jorge Héctor, correo electrónico: collaguazo.hector09@gmail.com o a su número celular 0981791219.