

**UNIVERSIDAD POLITÉCNICA SALESIANA  
SEDE QUITO**

**CARRERA:  
INGENIERÍA ELECTRÓNICA**

**Trabajo de titulación previo a la obtención del título de:  
INGENIERO ELECTRÓNICO**

**TEMA:  
DESARROLLO DE UN PROTOTIPO PARA LA COMUNICACIÓN DE  
PACIENTES CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)**

**AUTOR:  
LUIS CARLOS COLLAGUAZO CHICAIZA**

**TUTOR:  
EDUARDO RODRIGO TORRES SANTOS**

**Quito, diciembre del 2018**

## **CESIÓN DE DERECHOS DE AUTOR**

Yo, Luis Carlos Collaguazo Chicaiza, con documento de identificación N° 180394989-8, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación intitulado “DESARROLLO DE UN PROTOTIPO PARA LA COMUNICACIÓN DE PACIENTES CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)”, mismo que ha sido desarrollado para optar por el título de Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



---

Luis Carlos Collaguazo Chicaiza  
C.I. 180394989-8

Quito, diciembre 2018.

## **DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR**

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el Proyecto Técnico, “DESARROLLO DE UN PROTOTIPO PARA LA COMUNICACIÓN DE PACIENTES CON ESCLEROSIS LATERAL AMIOTRÓFICA (ELA)”, realizado por Luis Carlos Collaguazo Chicaiza, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerado como trabajo final de titulación.

Quito, diciembre 2018.

A handwritten signature in blue ink, consisting of a large, sweeping loop at the top and several smaller, more intricate strokes below it.

---

Eduardo Rodrigo Torres Santos

C.I. 171365460-4

## **DEDICATORIA**

A Dios, por darme la oportunidad de vivir y por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente, por haberme dado salud para lograr mis objetivos y por permitirme llegar hasta este momento tan importante de mi formación profesional.

A mi madre María Chicaiza, por ser el pilar más importante de mi vida, por sus consejos, por la motivación, por el esfuerzo y sacrificio realizado durante el trayecto de mi vida universitaria... Mamá, gracias por darme la educación la cual me servirá para mi futuro.

A mi padre Jorge Collaguazo, quien es un ejemplo para mí y la persona que admiro, siendo a la vez mi meta a superar.

A mis hermanos quienes me apoyaron en todo momento... Los quiero mucho.

*Luis Carlos*

## **AGRADECIMIENTO**

En primer lugar doy, infinitamente, gracias a Dios, por la vida de mis padres; también porque cada día bendice mi vida con la hermosa oportunidad de estar y disfrutar de los momentos inolvidables con las personas que más amo, por darme fuerzas para superar los obstáculos y dificultades a lo largo de toda mi vida.

Agradezco a mis padres por ser mi pilar fundamental y haberme apoyado, incondicionalmente, en todo momento, porque a través de su amor, esfuerzo y paciencia me ayudaron a culminar mi carrera universitaria.

A mis hermanos: Marcial, Esther, Héctor y a mí cuñado Raúl quienes con su ayuda, cariño y comprensión han sido parte fundamental de mi vida.

A mi Tutor de Trabajo de Titulación, Ingeniero Eduardo Torres, por toda la colaboración brindada, durante la elaboración de este proyecto.

A la Universidad Politécnica Salesiana, en especial, a la Carrera de Ingeniería Electrónica y a sus docentes, quienes me han dado la pauta para mi formación profesional.

A mi prometida Liliana, por ser el apoyo incondicional en mi vida, que con su amor y respaldo me ayuda a alcanzar mis objetivos.

Finalmente, mis sinceros agradecimientos a toda mi familia y amigos quienes me apoyaron en el transcurso de mi vida universitaria.

*Luis Carlos*

## ÍNDICE DE CONTENIDO

Cesión de derechos de autor.....	i
Declaratoria de coautoría del docente tutor .....	ii
Dedicatoria .....	iii
Agradecimiento .....	iv
Índice de contenido .....	v
Índice de figura .....	viii
Índice de tablas.....	x
Resumen.....	xi
Abstract .....	xii
Introducción .....	xiii
<b>CAPÍTULO 1.....</b>	<b>1</b>
<b>ANTECEDENTES .....</b>	<b>1</b>
1.1 Tema.....	1
1.2 Justificación.....	1
1.3 Planteamiento del problema.....	1
1.4 Objetivos .....	2
1.4.1 Objetivo General .....	2
1.4.2 Objetivos Específicos.....	2
1.5 Alcance.....	3
<b>CAPÍTULO 2.....</b>	<b>4</b>
<b>MARCO TEÓRICO .....</b>	<b>4</b>
2.1 Esclerosis Lateral Amiotrófica.....	4
2.1.1 Causas de ELA.....	5
2.1.2 Diagnósticos de ELA .....	5
2.1.3 Síntomas de ELA .....	6
2.1.4 Etapas de la enfermedad de ELA .....	6
2.1.5 Tratamiento para la enfermedad de ELA .....	7
2.2 Tecnología de Asistencia .....	7
2.2.1 Clasificación de las Tecnologías de Asistencia según su nivel.....	8
2.2.2 Tecnologías de Asistencia para personas con discapacidad .....	8
2.3 Etapas de sistema electrónico .....	10
2.3.1 Dispositivos de entrada .....	11

2.3.2 Dispositivos de proceso .....	11
2.3.3 Dispositivos de salida.....	12
2.3.4 Dispositivos de Entrada/Salida .....	13
2.4 Software libre .....	13
2.5 Teclada virtual.....	14
2.5.1 Texto predictivo .....	14
2.5.2 Tipos de teclados virtuales con textos predictivos.....	14
2.6 Tecnología Push .....	15
2.6.1 Notificación Push.....	15
<b>CAPÍTULO 3.....</b>	<b>17</b>
<b>DISEÑO E IMPLEMENTACIÓN .....</b>	<b>17</b>
3.1 Análisis de hardware .....	17
3.1.1 Single Board Computer (SBC) .....	17
3.1.2 Cámara .....	19
3.1.3 Pantalla.....	20
3.1.4 Parlantes .....	22
3.2 Análisis de software .....	23
3.2.1 Sistema operativo .....	23
3.2.2 Lenguaje de programación.....	24
3.3 Representación gráfica del algoritmo del prototipo .....	24
3.4 Implementación de hardware .....	27
3.5 Implementación de software .....	30
3.5.1 Instalación de librerías y aplicaciones.....	31
3.5.2 Desarrollo de la interfaz gráfica.....	32
<b>CAPÍTULO 4.....</b>	<b>36</b>
<b>PRUEBAS Y RESULTADOS .....</b>	<b>36</b>
4.1 Funcionamiento del prototipo final.....	36
4.2 Seguimiento facial.....	37
4.3 Interfaz gráfica .....	38
4.3.1 Teclado virtual .....	38
4.3.2 Mensajes predefinidos.....	39
4.3.3 Notificación Push.....	40
4.3.4 Reproductor multimedia.....	40
4.4 Validación del prototipo.....	41

4.5 Análisis de resultados.....	48
4.6 Análisis económico .....	49
4.6.1 Costo de hardware.....	49
4.6.2 Costo de diseño .....	50
4.6.3 Costo total del prototipo.....	50
4.7 Conclusiones y Recomendaciones .....	54
4.7.1 Conclusiones .....	54
4.7.2 Recomendaciones.....	55
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>56</b>
<b>ANEXOS</b>	



## ÍNDICE DE FIGURA

Figura 2. 1: Esclerosis Lateral Amiotrófica (ELA).....	5
Figura 2. 2: Dispositivo de la empresa Tobii .....	9
Figura 2. 3: Dispositivo Sip/ Puff Switch .....	9
Figura 2. 4: Dispositivo Irisbond .....	10
Figura 2. 5: Sistema electrónico.....	10
Figura 2. 6: Dispositivos de entrada del sistema electrónico .....	11
Figura 2. 7: Dispositivos de proceso del sistema electrónico .....	12
Figura 2. 8: Dispositivos de salida del sistema electrónico .....	12
Figura 2. 9: Dispositivos de Entrada/Salida del sistema electrónico .....	13
Figura 2. 10: Envío de notificación Push .....	16
Figura 3. 1: Etapas empleadas para el desarrollo del prototipo .....	17
Figura 3. 2: Raspberry Pi 3 .....	19
Figura 3. 3: Módulo de la cámara RPI .....	20
Figura 3. 4: Pantalla Touch Creen de 10.1” .....	21
Figura 3. 5: Parlantes para la reproducción de audio .....	22
Figura 3. 6: Diagrama de flujo para el desarrollo del software.....	25
Figura 3. 7: Diagrama de flujo para el uso del prototipo .....	26
Figura 3. 8: Diagrama esquemático .....	27
Figura 3. 9: Diseño de la estructura para el prototipo en AutoCAD.....	28
Figura 3. 10: Estructura para el prototipo .....	28
Figura 3. 11: Conexión y ubicación de los elementos .....	29
Figura 3. 12: Prototipo final de comunicación.....	29
Figura 3. 13: Carga de Ubuntu Mate al micro SD .....	30
Figura 3. 14: Entorno de Ubuntu Mate .....	30
Figura 3. 15: Ventana de configuración de periféricos de RPI.....	31
Figura 3. 16: Instalación de Eviacam.....	31
Figura 3. 17: Código empleado para la creación de la ventana .....	32
Figura 3. 18: Formato para crear el diccionario de palabras.....	32
Figura 3. 19: Canal y ApiKey creado en Pushetta .....	34
Figura 3. 20: Suscripción en el teléfono Android .....	34
Figura 3. 21: Código empleado para el envío del mensaje de emergencia.....	35
Figura 4. 1: Prototipo final de comunicación.....	36

Figura 4. 2: Seguimiento facial .....	37
Figura 4. 3: Menú principal para el paciente.....	38
Figura 4. 4: Teclado virtual para el paciente.....	39
Figura 4. 5: Ventana de mensajes predefinidos .....	39
Figura 4. 6: Mensaje de notificación.....	40
Figura 4. 7: Reproductor SMPlayer .....	40
Figura 4. 8: Representación gráfica del resultado de la pregunta 1 .....	46
Figura 4. 9: Representación gráfica del resultado de la pregunta 2 .....	46
Figura 4. 10 Representación gráfica del resultado de la pregunta 3 .....	47
Figura 4. 11: Representación gráfica del resultado de la pregunta 4 .....	47
Figura 4. 12: Representación gráfica del resultado de la pregunta 5 .....	48

## ÍNDICE DE TABLAS

Tabla 2. 1: Libertades esenciales de los usuarios.....	14
Tabla 3. 1: Tabla comparativa de los ordenadores de placa reducida.....	18
Tabla 3. 2: Tabla comparativa de las cámaras para el seguimiento facial .....	20
Tabla 3. 3: Comparativa de las pantallas para RPI 3 .....	21
Tabla 3. 4: Comparativa de sistemas operativos para la Raspberry.....	23
Tabla 4. 1: Análisis del tiempo del primer mensaje.....	41
Tabla 4. 2: Análisis del tiempo del segundo mensaje .....	42
Tabla 4. 3: Análisis del tiempo del tercer mensaje .....	43
Tabla 4. 4: Análisis del tiempo del cuarto mensaje .....	44
Tabla 4. 5: Análisis del tiempo del quinto mensaje .....	45
Tabla 4. 6: Promedio del tiempo por palabra.....	49
Tabla 4. 7: Costo de hardware.....	49
Tabla 4. 8: Costo total del prototipo.....	50
Tabla 4. 9: Comparativa del prototipo desarrollado y el dispositivo EyeMobile Mini .....	51
Tabla 4. 10: Comparativa del prototipo desarrollado y el dispositivo Tobii I15+ Plus .....	53

## RESUMEN

El presente proyecto de titulación detalla el desarrollo e implementación de un prototipo para la comunicación de pacientes con Esclerosis Lateral Amiotrófica (ELA). Esta rara enfermedad se presenta, generalmente, en personas mayores de 50 años, aunque en casos minoritarios aparece antes de esta edad. Es una enfermedad neurodegenerativa progresiva que destruye a las neuronas motoras del cerebro y la médula espinal, donde las neuronas dejan de enviar impulsos eléctricos a los músculos causando la inmovilidad de las piernas, los brazos, e incluso la lengua, perdiendo casi en su totalidad el movimiento del cuerpo, lo cual limita la comunicación con las personas de su alrededor.

El prototipo tiene una interfaz gráfica que consta de cuatro botones, los cuales permiten acceder a la simulación del teclado virtual, envío de notificaciones, frases predefinidas y reproductor multimedia, respectivamente. A través del teclado virtual el paciente puede escribir el mensaje mediante el movimiento de la cabeza, este mensaje es convertido en voz mediante la tecnología TTS, para mayor rapidez en la escritura se tiene un diccionario con las palabras más empleadas en el lenguaje español. Además, con el botón: *envío de notificaciones* se puede enviar un mensaje de emergencia hacia un dispositivo Android.

El dispositivo consta de un módulo de la cámara conectado al puerto CSI de la tarjeta Raspberry Pi 3 y utilizando la aplicación de eViacam se controla el cursor del mouse a través del movimiento de la cabeza. La interfaz gráfica se desarrolló utilizando el lenguaje de programación Python y la librería Tkinter.

## **ABSTRACT**

The present project of qualification details the development and implementation of a prototype for the communication of patients with Amyotrophic Lateral Sclerosis (ALS), this rare disease usually arises in people older than 50 years, although in minority cases it appears before this age, it is a neurodegenerative disease progressive that destroys the motor neurons of the brain and spinal cord, where the neurons stop sending electrical impulses to the muscles causing the immobility of the legs, the arms, even the tongue, losing almost the entire movement of the body, which limits communication with the people around you.

The prototype has a graphic interface that consists of four buttons, which allow access to the simulation of the virtual keyboard, shipping notifications, predefined phrases and media player respectively. Through the virtual keyboard the patient can write the message through the movement of the head, this message is converted into voice using TTS technology, a dictionary with the words most used in the Spanish language is for writing faster. In addition, to the button sending notifications can send a message of emergency to an Android device

The device consists of a camera module connected to the CSI port of the Raspberry Pi 3 card and using the EVIACAM application, the mouse cursor is controlled through the movement of the head. The graphic interface was developed using the Python programming language and the Tkinter library.

## INTRODUCCIÓN

El desarrollo de prototipo para la comunicación de pacientes con Esclerosis Lateral Amiotrófica (ELA) tiene como propósito dar una alternativa de comunicación a las personas que padecen esta enfermedad. Las personas que sufren de esta terrible enfermedad, tienen que soportar la parálisis de sus extremidades e incluso la parálisis total del cuerpo con excepción de la movilidad al menos de la cabeza en la etapa denominada tardía.

En el capítulo uno, se detalla todo lo referente al tema, la justificación, el planteamiento del problema y los objetivos general y específicos del proyecto de titulación.

El capítulo dos, contiene la parte teórica que conforma todo el proyecto, partiendo de un análisis completo de la enfermedad de ELA, conociendo sus etapas, investigando los diferentes dispositivos electrónicos de ayuda que existen para esta enfermedad y estudiando el sistema electrónico para el diseño y desarrollo del prototipo.

En el capítulo tres, se describe de manera detallada y por partes el desarrollo e implementación del prototipo para la comunicación de pacientes con ELA.

En el capítulo cuatro, se presenta los resultados de las pruebas de funcionamiento y un análisis comparativo del prototipo con dispositivos comerciales para determinar su costo-beneficio. También se muestran las conclusiones y recomendaciones obtenidas durante el desarrollo del proyecto de titulación.

# **CAPÍTULO 1**

## **ANTECEDENTES**

En este capítulo se detalla todo lo referente al tema, la justificación, el planteamiento del problema, el objetivo general, los objetivos específicos y el alcance referente al proyecto a desarrollarse.

### **1.1 Tema**

Desarrollo de un prototipo para la comunicación de pacientes con Esclerosis Lateral Amiotrófica (ELA).

### **1.2 Justificación**

La enfermedad de ELA en el Ecuador se ha ido incrementado, considerablemente, en los últimos años. Razón por la cual es importante la realización del proyecto que permita la comunicación de personas que padecen esta enfermedad, debido a que pierden progresivamente el movimiento de su cuerpo, incluso de la lengua y se les limita la comunicación con las personas de su alrededor, y así ayudar a que su autoestima se eleve y puedan comunicarse por sí mismos. (Mosquera, 2016)

Existen prototipos relacionados con equipo biomédico desarrollados en proyectos de titulación que justifican la existencia en el Ecuador de la tecnología requerida para incursionar en este tema.

### **1.3 Planteamiento del problema**

La sociedad actual atraviesa un sinnúmero de enfermedades, que atacan e incluso acaban con la vida de las personas, como es el caso de ELA, también conocido como la enfermedad del científico Stephen Hawking. Esta terrible enfermedad ataca a las neuronas motoras del cerebro. Las neuronas motoras son las que envían mensajes desde el cerebro a los músculos de todo el cuerpo. A medida que las neuronas motoras mueren, la persona pierde progresivamente la capacidad de caminar, hablar, tragar e incluso de respirar. El individuo que padece de esta enfermedad está limitado de poder comunicarse de forma natural con las personas de su alrededor. (Project ALS, 2018)

De acuerdo a la Asociación Española de Esclerosis Lateral Amiotrófica (ADELA), la enfermedad de ELA afecta, especialmente, a personas de edades comprendidas entre los 40 y 70 años, aunque también se puede presentar fácilmente en los jóvenes. Esta enfermedad es más frecuente en varones. En España se estiman unas 4000 personas afectadas y 350.000 a nivel mundial. En el año se diagnostican unos 120.000 casos nuevos en todo el mundo. (ADELA, 2014)

De acuerdo a la investigación realizada por la Fundación Ecuatoriana de Esclerosis Múltiple (FUMDEM) en el Ecuador, ha comprobado que en el año 2014 la enfermedad de ELA era casi desconocida ya que se presentaron únicamente 5 casos, pero en el año 2016 se ha presentado 50 casos de ELA. Esto es preocupante porque se puede verificar que la enfermedad va aumentando considerablemente; sin embargo, en el Ecuador no se producen dispositivos de asistencia, que faciliten la comunicación de los pacientes con ELA. Los precios para la adquisición de estos dispositivos que son implementados en el extranjero son demasiados costosos, ya que son realizados por grandes empresas, volviéndose inaccesibles para personas de recursos económicos medios o bajos. (Mosquera, 2016)

## **1.4 Objetivos**

### **1.4.1 Objetivo General**

Desarrollar un prototipo para la comunicación de pacientes con Esclerosis Lateral Amiotrófica (ELA) mediante single board computer (SBC) y software libre.

### **1.4.2 Objetivos Específicos**

- Analizar la etapa tardía que sufren los enfermos con ELA para determinar los parámetros de diseño de la tecnología de asistencia requerida.
- Diseñar el prototipo de asistencia para la comunicación del paciente con ELA utilizando ordenador de placa reducida y componentes electrónicos auxiliares.
- Desarrollar una interfaz gráfica para la comunicación a través de la selección de palabras y sintetizador de voz.



- Comparar el prototipo con tecnologías comerciales para determinar el beneficio económico.

### **1.5 Alcance**

Mediante el desarrollo del presente proyecto, se pretende ayudar sustancialmente la calidad de vida de las personas con discapacidades, especialmente a personas con discapacidades motoras como el caso de los pacientes con ELA.

El prototipo para la comunicación tiene una interfaz gráfica que consta de un menú principal y constituye de cuatro botones, los cuales permiten acceder a la simulación del teclado virtual, envío de notificaciones, frases predefinidas y reproductor multimedia respectivamente. A través del teclado virtual el paciente podrá escribir el mensaje deseado, dicho mensaje será convertido en voz usando la tecnología TTS y para mayor rapidez en la escritura se tiene un diccionario con las palabras más empleadas en el lenguaje español. También brinda la facilidad de enviar una notificación de emergencia hacia un teléfono Android.

Cabe aclarar que el prototipo descrito no es portable y para su funcionamiento se debe conectar a un punto de red eléctrico, además se manifiesta que no incluye una silla y se hace uso de la conexión WIFI no incluida únicamente para el envío de notificaciones.

## **CAPÍTULO 2**

### **MARCO TEÓRICO**

En este capítulo se hace una breve descripción sobre los conceptos de la enfermedad de ELA, sus causas, síntomas, etapas y de los componentes que intervienen en el proyecto. La descripción teórica está basada en los libros técnicos, las monografías, los artículos académicos y documentación de las páginas Web.

#### **2.1 Esclerosis Lateral Amiotrófica**

La enfermedad de Esclerosis Lateral Amiotrófica afecta a las células nerviosas del cerebro y la médula espinal, así dejando de enviar los impulsos eléctricos a los músculos y evitando el funcionamiento normal, debido a que este mal es neurodegenerativo progresivo que provoca debilidad muscular e incapacidad de movimiento en las extremidades e incluso afectando en la respiración. Los enfermos que padecen de este mal en las etapas finales pueden quedar totalmente paralizados. (ALS ASSOCIATION, 2018)

La enfermedad de ELA no afecta a los sentidos de: ver, oír, gustar, oler y tocar, normalmente, tampoco afecta a la mente, los músculos de los ojos, el corazón, vejiga, intestino, o músculos sexuales. No presenta la posibilidad de que la enfermedad de ELA sea contagiosa. (Bossa, Abarca, Torres, Ramírez, & Garcia, s.f)

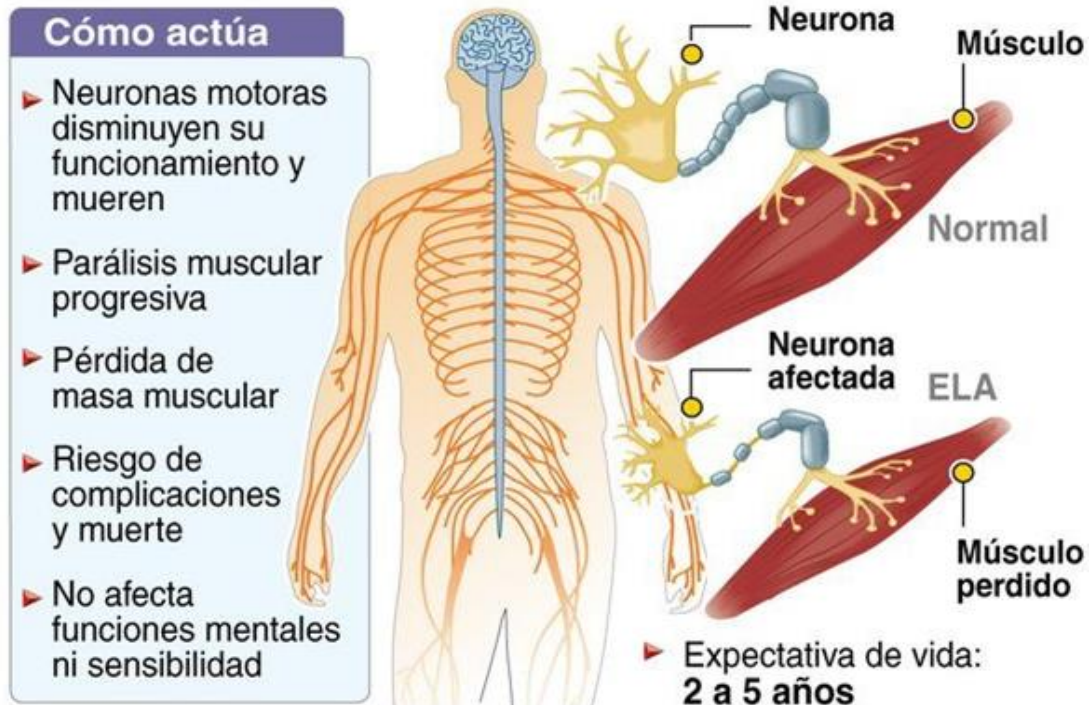
Las neuronas motoras son conocidas como células nerviosas y se dividen en dos: las neuronas motoras inferiores que se encargan de controlar los músculos de las piernas y pies, y las neuronas motoras superiores que se encargan de controlar los músculos de los brazos, manos y dedos. Además, el hablar, el deglutir y el tragar están controlados por las moto-neuronas que se encuentran en el tallo del cerebro. (Bossa, Abarca, Torres, Ramírez, & Garcia, s.f)

En la Figura 2.1 se muestra como la enfermedad de ELA daña las células nerviosas motoras causando que los músculos se debiliten.

Figura 2. 1: Esclerosis Lateral Amiotrófica (ELA)

# Esclerosis Lateral Amiotrófica (ELA)

La enfermedad provoca una gran discapacidad motora



Esclerosis Lateral Amiotrófica (ELA), Fuente: (Consejos de tu farmacéutico, 2016)

## 2.1.1 Causas de ELA

Se considera que la enfermedad de ELA es de carácter hereditario en un 5 a 10 por ciento de los pacientes, mientras que el resto no tiene causa conocida ya que se presenta de forma esporádica. Se puede considerar causas como son los factores genéticos, factores ambientales y factores relacionados con la edad, no se ha confirmado ninguna otra causa cuando este tema se ha estudiado con precisión científica. (MAYO CLINIC, 2017)

## 2.1.2 Diagnósticos de ELA

La enfermedad de ELA se diagnostica en base a una historia clínica detallada de los signos y síntomas observados por un médico durante el examen físico, incluyendo otras pruebas que ayuden en la determinación de ELA y no confundir con otras enfermedades. Para evaluar los síntomas como debilidad, atrofia o espasticidad muscular. El médico estudia la historia clínica del paciente y efectúa

el examen neurológico para determinar si las extremidades están empeorando progresivamente. (National Institutes of Health, 2018)

La de enfermedad de ELA en las etapas iniciales pueden confundir con otras enfermedades que son tratables. Es más probable diagnosticar ELA cuando la enfermedad está avanzada, es decir, cuando varias partes del cuerpo están afectadas. (National Institutes of Health, 2018)

### **2.1.3 Síntomas de ELA**

Al aparecer la ELA, los síntomas pueden ser tan leves que a menudo pasan desapercibidos. Pero con el avance de la enfermedad los síntomas podrían incluir lo siguiente:

- a. Dificultad para realizar actividades físicas cotidianas.
- b. Tropiezos y caídas
- c. Debilidad en las extremidades
- d. Balbuceo o dificultad al tragar
- e. Calambres musculares y fasciculaciones en las extremidades e incluso en la lengua
- f. Dificultad para mantener una postura correcta del cuerpo

Los músculos comienzan a debilitarse progresivamente mientras avanza la enfermedad debido a la pérdida de las neuronas motoras del cerebro. La enfermedad de ELA comienza en las extremidades como son las manos, pies o miembros inferiores y, luego, se extiende a otras partes del cuerpo. (MAYO CLINIC, 2017)

### **2.1.4 Etapas de la enfermedad de ELA**

La ELA es una condición que limita la vida de las personas en forma progresiva, afectando los músculos del cuerpo, en consecuencia, la movilidad de las extremidades que varía de acuerdo a la condición física de cada persona, sin embargo, las etapas que generalmente se presentan en esta enfermedad son cuatro: (Wang, W.Y.Su, Han, Lin, & Hsu, 2015)

- **Etapa temprana:** Los pacientes comienzan a notar una leve debilidad muscular. Las extremidades del cuerpo todavía están en funcionamiento, pero el nivel de destreza está bajando.
- **Etapa media:** Afecta la debilidad muscular progresiva en los brazos y piernas. En esta etapa ya es difícil mover brazos, piernas y el cuerpo.
- **Etapa tardía:** La debilidad muscular y la atrofia se extienden a otras partes del cuerpo. Los brazos y las piernas son apenas movidos. Los pacientes son capaces de comunicarse por expresiones faciales emocionales y el movimiento de la cabeza.
- **Etapa final:** En esta etapa los músculos más voluntarios están sin movilidad, el movimiento voluntario de los ojos sigue siendo funcional, pero la velocidad de movimiento del ojo puede ser lento.

### **2.1.5 Tratamiento para la enfermedad de ELA**

La enfermedad de ELA no tiene cura, únicamente existen tratamientos para retrasar la progresión y controlar los dolores musculares que esta ocasiona, como es el caso del fármaco riluzol. Sin embargo, es costoso pero efectivo. Los medicamentos recetados incluyen: (UCSF Health, 2018)

- Los medicamentos como el baclofeno o el diazepam ayudan a controlar la espasticidad.
- La gabapentina ayuda a controlar el dolor.
- Trihexyphenidyl o amitriptyline ayudan a los pacientes a tragar saliva.

Los tratamientos para ayudar a que los músculos funcionen lo mejor posible y mejorar su salud general, es con la fisioterapia, la rehabilitación, el uso de los aparatos ortopédicos o una silla de ruedas. Además, los pacientes que se ven afectados en el habla pueden hacer terapias con sistemas de comunicación basados en computadoras y con la tecnología sintetizador de voz. (UCSF Health, 2018)

### **2.2 Tecnología de Asistencia**

La tecnología de asistencia (AT) es un dispositivo o sistema electrónico que minimiza o elimina las barreras que limitan el aprendizaje y comunicación en aquellas personas que sufren de alguna discapacidad física. (Univisión-Noticias, 2014)

Gracias a la tecnología de asistencia, las personas con discapacidades físicas pueden llevar una vida productiva, independiente, digna y formar parte en la sociedad. Las personas que no tienen el acceso a esta tecnología suelen verse aisladas, lo que hace más terrible las consecuencias de una discapacidad para la propia persona, su familia y la sociedad. Únicamente una de cada 10 personas que sufren de discapacidades físicas tiene acceso a tecnología de asistencia, debido a su elevado costo, y su escasa existencia en el mercado. (Organización Mundial de la Salud, 2018)

### **2.2.1 Clasificación de las Tecnologías de Asistencia según su nivel**

- **De baja tecnología:** Utilizan herramientas de adaptación simples como: bastones, andadores, barras de sujeción en el baño, utensilios de cocina especiales, materiales de lectura de letra grande, aparatos ortopédicos para los pies, brazos y manos. De acuerdo a los estudios realizados se considera que los dispositivos de movilidad son los más utilizados en este grupo de tecnología de asistencia. (Grey & Cook, s.f)
- **De alta tecnología:** Se hace uso de los productos y equipos de gran complejidad tecnológica como son: comunicadores personales, sillas de ruedas auto- guiadas, modificación de vehículos, sistemas de comunicación que están programados para emitir voz, prótesis con articulaciones para mover en varios planos y lectores de pantallas gráficas, los cuales son inaccesibles para personas de recursos económicos medios o bajos. (Grey & Cook, s.f)

### **2.2.2 Tecnologías de Asistencia para personas con discapacidad**

En los últimos años, la tecnología ha estado evolucionando considerablemente ayudando a las personas que padecen de algún tipo de discapacidad lleven una vida productiva e independiente gracias a los dispositivos y softwares existentes para tal fin. A continuación, se muestra algunos de ejemplos.

#### **- Tobii:**

Es una empresa Sueca líder mundial en la tecnología de asistencia que desarrolla dispositivos de comunicación estructuradas para ayudar en las necesidades específicas de las personas con las enfermedades de: parálisis cerebral, síndrome de Rett,

apoplejía, ELA y otras enfermedades que limitan la comunicación. En la Figura 2.2 se puede apreciar el dispositivo creado por la empresa Tobii. (Tobii, 2015)

Figura 2. 2: Dispositivo de la empresa Tobii



Dispositivo para la comunicación de la empresa Tobii, Fuente: (Tobii Dynavox, 2016)

#### - **Sip/ Puff Switch:**

Esta tecnología de asistencia es desarrollada para las personas con discapacidad motriz, que a través de un tubo conectado a la boca reconoce las respiraciones y movimientos de la lengua como señales para el control de diferentes dispositivos, como el manejar una silla de ruedas hasta navegar por internet. Además, puede tener el acceso a los dispositivos más sofisticados mediante softwares especializados. En la Figura 2.3 se muestra el dispositivo Sip/ Puff Switch. (Ortega, 2018)

Figura 2. 3: Dispositivo Sip/ Puff Switch



Dispositivo Sip/ Puff Switch, Fuente: (Origen Instruments, s.f)

### - Irisbond:

El sistema Irisbond se trata de una tecnología de comunicación alternativa basada en visión artificial, el cual permite controlar un ordenador con el movimiento de los ojos, y por lo tanto acceder a todas las aplicaciones que tiene un ordenador. Este dispositivo es una opción muy útil para la comunicación de aquellas personas que padecen de alguna discapacidad motriz (ELA), que al no poder hablar ni escribir están limitados de poder comunicarse con las personas que les rodean. En la Figura 2.4 se puede apreciar la tecnología Irisbond. (Llorens, 2015)

Figura 2. 4: Dispositivo Irisbond



Dispositivo Irisbond, Fuente: (ELA Andalucía, 2017)

### 2.3 Etapas del sistema electrónico

Los sistemas electrónicos son conjuntos de circuitos que trabajan con señales eléctricas y las tratan para la ejecución de una determinada función. Constan de una etapa de entrada, de una etapa de proceso o control y de una etapa de salida como se observa en la Figura 2.5. (Leira & Gómez, 2016)

Figura 2. 5: Sistema electrónico



Etapas del sistema electrónico, Fuente: (Departamento de Tecnología, s.f)



### 2.3.1 Dispositivos de entrada

Generan una señal eléctrica a partir de una señal exterior de otro tipo como son las señales de: temperatura, humedad, presión, movimiento, nivel, luz, pulsación en un teclado, etc. En la Figura 2.6 se ilustra algunos dispositivos de entrada.

(Departamento de Tecnología, s.f)

- Sensores
- Teclados
- Cámaras IP
- Pulsadores, etc.

Figura 2. 6: Dispositivos de entrada del sistema electrónico



Dispositivos de entrada del sistema electrónico, Elaborado por: Luis Collaguazo

### 2.3.2 Dispositivos de proceso

El dispositivo de proceso es un sistema que obtiene la señal de entrada, la procesa y genera una señal de salida. Dicho procesamiento radica en la realización de operaciones lógicas, cálculos matemáticos de diversos tipos o conversiones de varios niveles de complejidad, como es el caso de la implementación de algoritmos de regulación y control. En la Figura 2.7 se ilustra algunos dispositivos de proceso.

(Municipalidad de San Cayetano, 2018)

- Microcontroladores
- PLCs
- Computadoras
- SBCs, etc.

Figura 2. 7: Dispositivos de proceso del sistema electrónico



Dispositivos de proceso del sistema electrónico, Elaborador por: Luis Collaguazo

### 2.3.3 Dispositivos de salida

Se denominan dispositivos de salida a los transductores como: actuadores e indicadores. Los actuadores convierten una señal eléctrica en otra forma de energía, generalmente mecánica y los indicadores permiten la visualización. En la Figura 2.8 se ilustra algunos dispositivos de salida. (Municipalidad de San Cayetano, 2018)

- Motores Eléctricos
- Lámparas
- Altavoces
- Visualizadores (Displays), etc.

Figura 2. 8: Dispositivos de salida del sistema electrónico



Dispositivos de salida del sistema electrónico, Elaborado por: Luis Collaguazo

### 2.3.4 Dispositivos de Entrada/Salida

Son aquellos dispositivos que se pueden operar de dos formas, como entrada o como salida. Generalmente se les conoce como periféricos de *Entrada/Salida* debido a que permite el ingreso de datos y muestra la información de los datos. En la Figura 2.9 se ilustra algunos dispositivos de Entrada/Salida. (Hernandez, 2015)

- Unidades de almacenamiento: CD, DVD, Disco duro, Pendrive ESB
- Modem
- Router
- Pantalla Táctil, etc.

Figura 2. 9: Dispositivos de *Entrada/Salida* del sistema electrónico



Dispositivos de Entrada/Salida del sistema electrónico, Elaborado por: Luis Collaguazo

### 2.4 Software libre

Software libre respeta la libertad de los usuarios y la comunidad. Esto hace referencia a la capacidad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software que está en sus manos. Es decir, el software libre es un asunto de libertad, no de precio; esto significa que el usuario tiene una gran ayuda debido a que el código fuente está a la disposición del usuario para realizar las adecuaciones, modificaciones y hacer mejoras de acuerdo a sus requerimientos particulares o generales de uso. El software libre incluye las 4 libertades de los usuarios como se muestra en la Tabla 2.1. (Illapa, 2017)

Tabla 2. 1: Libertades esenciales de los usuarios

LIBERTADES DE LOS USUARIOS	
Libertad 0	Libertad de ejecutar el programa de acuerdo a su propósito.
Libertad 1	Libertad de estudiar cómo funciona el programa, y modificarlo de acuerdo a sus necesidades de uso (Es indispensable tener acceso al código fuente).
Libertad 2	Libertad de redistribuir copias, y ayudar a otras personas.
Libertad 3	La libertad de modificar, con la finalidad de ofrecer mejoras a toda la comunidad (Es indispensable tener acceso al código fuente).

Libertades esenciales de los usuarios, Fuente: (Free Software Foundation, 2018)

## 2.5 Teclada virtual

Los teclados virtuales son programas que emiten las funciones del teclado físico convencional. Al realizar la acción del *clic* sobre un botón del teclado virtual que se muestra en la pantalla se obtiene el mismo efecto que el teclado convencional. (Ministerio de Educación y Ciencia, s.f)

El uso del teclado virtual es útil en personas con discapacidades de movilidad y para la ayuda en la mecanografía, que no pueden o no saben manejar el teclado convencional. Además, algunos teclados virtuales tienen sistemas de predicción de palabras, el cual ahorra pulsaciones y agiliza la escritura. (Ministerio de Educación y Ciencia, s.f)

### 2.5.1 Texto predictivo

Los predictores de palabras funcionan al hacer referencia a un diccionario con las palabras más utilizadas por los usuarios. El texto predictivo, es una tecnología de entrada de texto que permite formar las palabras deseadas al presionar una sola tecla por cada letra y de esta forma simplificar la escritura de mensajes. (ABC TECNOLOGÍA, 2014)

### 2.5.2 Tipos de teclados virtuales con textos predictivos

Existen varios tipos de teclados con métodos y estilos de textos predictivos, a continuación, se menciona dos de ellos:

- **T9:** Es la primera tecnología desarrollada para textos predictivos, Texto en nueve teclas. La pulsación de cada uno de las teclas introduce una letra. Esto se realiza mediante una serie de palabras que ofrece el diccionario. Sin embargo, esta tecnología no es perfecta, ya que una secuencia de teclas determinada puede corresponder a más de una palabra. Esta tecnología se desarrolló para brindar la facilidad a la hora de escribir mensajes. (ABC TECNOLOGÍA, 2014)
  
- **Swiftkey:** Es una de las tecnologías más recientes, utiliza el método de ingreso de palabras por medio de inteligencia artificial, el cual es capaz de predecir la palabra que el usuario tiene la intención de escribir. Este sistema «aprende» de anteriores mensajes. Además, esta tecnología es desarrollada para los dispositivos con pantalla táctil y es de fácil uso, de tal forma que para realizar la escritura el usuario solo debe deslizar el dedo y formar palabras. (ABC TECNOLOGÍA, 2014)

## **2.6 Tecnología Push**

Es una tecnología informática que se encarga de transmitir la información que los usuarios necesitan desde Internet. (Cáceres & Moyano, s.f, pág. 3)

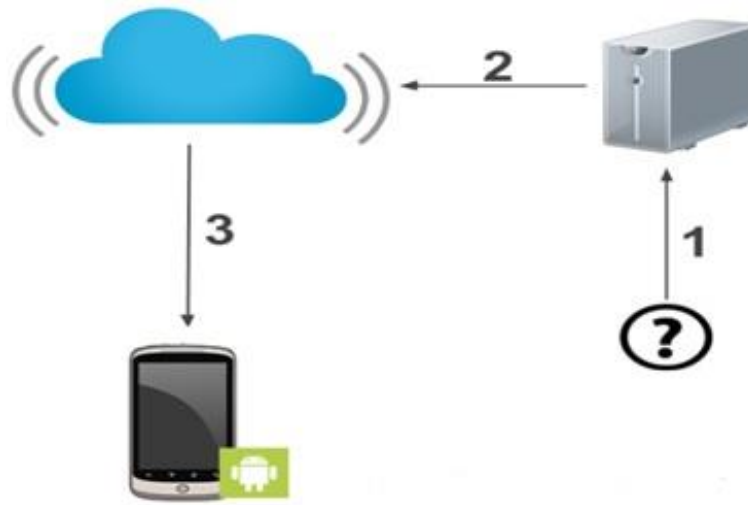
### **2.6.1 Notificación Push**

Una notificación Push es un mensaje enviado por un servidor a sus clientes que están "suscritos" a su canal. Las comunicaciones Push, únicamente son posibles en entornos cliente-servidor, en los que uno o muchos clientes se suscriben a un topic concreto de los ofrecidos por el servidor. En la Figura 2.10 se ilustra el envío de notificación Push. (Manumateos, 2013)

A continuación, se muestra algunos ejemplos de la notificación Push.

- Mailing profesional
- Envío masivo de SMS
- Notificaciones Push APP
- Notificaciones WEB Push

Figura 2. 10: Envío de notificación Push



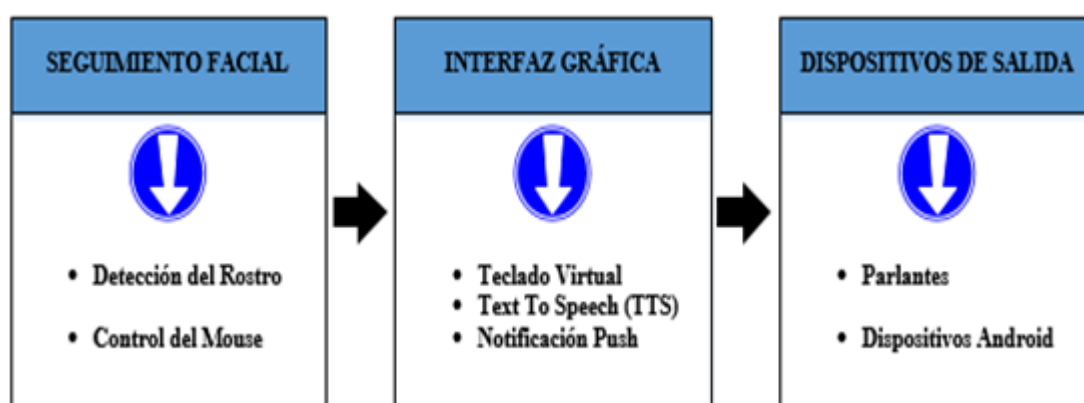
Envío de notificación Push, Elaborado por: Luis Collaguazo

## CAPÍTULO 3

### DISEÑO E IMPLEMENTACIÓN

En el presente capítulo se detalla el software y el hardware empleado para la elaboración del prototipo de comunicación para los pacientes con ELA, el cual es un sistema alternativo de comunicación y consta de las tres etapas principales. En la Figura 3.1 se muestran las etapas empleadas.

Figura 3. 1: Etapas empleadas para el desarrollo del prototipo



Etapas empleadas para el desarrollo del prototipo, Elaborado por: Luis Collaguazo

### 3.1 Análisis de hardware

Para el desarrollo del prototipo de comunicación previamente se realizó un análisis de los diferentes dispositivos electrónicos existentes en el mercado con la finalidad de garantizar una funcionalidad apropiada.

#### 3.1.1 Single Board Computer (SBC)

Entre los Single Board Computer o en español Ordenadores de Placas Reducidas existen diversas marcas como: BeagleBone Black, Odroid C2, Paspberry Pi y LattePanda, cada una con diferentes características técnicas en su interfaz, sistema operativo, procesador, almacenamiento, conectividad y el precio. En la Tabla 3.1 se muestra un estudio comparativo de cada uno de las placas.

Tabla 3. 1: Tabla comparativa de los ordenadores de placa reducida

Características	BeagleBone Black	Odroid C2	Raspberry Pi 3	LattePanda
Procesador	1GHz ARM AM335x	1.5 GHz Quad-Core ARM Cortex A53	1.2GHz 64 bits. Quad-Core ARMv8	1.8 GHz 64-Bit Quad-Core Intel Z8300
Memoria interna	4GB	Tarjeta SD	Tarjeta micro SD	32 GB 64 GB
RAM	512 MB	2 GB	1 GB	2 GB 4 GB
Conectividad	1 USB 2.0 Micro HDMI Ethernet	4 USB 2.0 1 USB OTG HDMI 2.0 Ethernet	4 USB 2.0 HDMI 1.4 Ethernet WIFI Bluetooth 4.1 Bluetooth (BLE)	1 USB 3.0 2 USB 2.0 WIFI Ethernet Bluetooth 4.0
Sistema Operativo	Linux Android	Linux Android	Linux Windows-IoT	Windows 10
Precio	90 \$	62 \$	35 \$	130 \$ 160 \$

Tabla comparativa de los ordenadores de placa reducida, Elaborado por: Luis Collaguazo

Después de analizar las características técnicas de las cuatro placas, se llegó a una conclusión que la Raspberry Pi 3 modelo B como se ilustra en la Figura 3.2, es el mini-ordenador que llena las expectativas para ser usado en el prototipo, debido a que tiene los periféricos ideales tales como: el conector HDMI, el puerto CSI, los conectores USB, conector de audio y WIFI, los cuales puede adaptar a los componentes como: cámara, teclado, mouse, pantalla, parlantes, etc. (Caiza & García, 2017) A continuación se detallan las especificaciones y requerimientos esperados.

- La conexión HDMI con la pantalla para visualizar la interfaz gráfica del paciente. (Torres Santos & Bucheli Naranjo, 2017)
- Puerto CSI para la conexión a la cámara, y mediante la aplicación Eviacam se hace un seguimiento facial y controlar el cursor del mouse.
- Puerto de conexión a los parlantes para la reproducción de audio.
- WIFI para acceder a internet y hacer en envío de notificaciones.



Las características que presenta este mini-ordenador son:

- 4 puertos USB 2.0
- 40 pines GPIO
- Puerto FULL HDMI 1.4
- Puerto Ethernet
- Interfaz de cámara (CSI)
- Interfaz de pantalla (DSI)
- Micro SD card
- Salida de audio de 3.5 mm
- Una CPU ARMv8 quad-core de 64 bits y 1.2 GHz
- Módulo de Wi-Fi b/g/n en la banda de 2.4GHz
- Bluetooth 4.1
- Bluetooth baja energía (BLE)
- Conector micro USB (Alimentación 5v y 2.5A)

Figura 3. 2: Raspberry Pi 3



Raspberry Pi 3, Elaborado por: Luis Collaguazo

### 3.1.2 Cámara

Una vez elegido la Raspberry Pi como el controlador central para utilizar en el prototipo, se conoce que esta placa ofrece dos opciones de conexión. La primera es el puerto USB donde se puede conectar cualquier webcam del mercado y la segunda es el módulo de la cámara RPI que se conecta mediante el puerto CSI. En la tabla 3.2 se muestra el estudio comparativo de las cámaras para el seguimiento facial.

Tabla 3. 2: Tabla comparativa de las cámaras para el seguimiento facial

Cámara	Resolución	Conector	Tamaño	Precio
Pi NoIR	5 MP	CSI	20x25x9 mm	33\$
Pi Cámara V1	5 MP	CSI	20x25x9 mm	28\$
Webcam	640x480 pixeles	USB	60x30x22 mm	19\$

Tabla comparativa de las cámaras seguimiento facial, Elaborado por: Luis Collaguazo

El módulo de la cámara Raspberry Pi, como se muestra en la Figura 3.3, cumple con las expectativas para ser usada debido a que captura fotos fijas y graba videos de alta definición, es económico, fácil de utilizar y presenta las siguientes características:

- **Resolución:** 5 megapíxeles
- **Tipo de conexión:** Puerto CSI no necesita previa para su comunicación.
- **Tamaño:** 25x20x9 mm lo cual facilita la ubicación en el prototipo.
- **Costo:** A pesar de ser más costoso que la webcam se escogió la Pi Cámara por que ofrece mejores características.

Figura 3. 3: Módulo de la cámara RPI



Módulo de la cámara RPI, Fuente: (gizmoj, 2015)

### 3.1.3 Pantalla

Para visualizar la interfaz gráfica que es manejada por el paciente se usó la pantalla Touch Screen. A continuación, en la Tabla 3.3 se muestra el estudio comparativo de las diferentes pantallas que son compatibles con la Raspberry PI 3.

Tabla 3. 3: Comparativa de las pantallas para RPI 3

<b>Descripción Técnica</b>	<b>Pantalla 5"</b>	<b>Pantalla 7"</b>	<b>Pantalla 10.1"</b>
Resolución	800 x 480	1024 x 600	1280 x 800
Sistema operativo	Soporta todo lo relacionado a RPI	Soporta todo lo relacionado a RPI	Soporta todo lo relacionado a RPI
Pantalla Táctil	SPI vía RPI	IPS screen, full view	IPS screen, full view
Dimensión	120mm x 74mm x 7 mm	210mm x 120mm x 8mm	265mm x 187mm x 1mm
Precio	\$ 39	\$ 58	\$132

Estudio comparativo de las pantallas para la RPI, Elaborado por: Luis Collaguazo

Después del estudio comparativo expuesto en la Tabla 3.3, se optó por la pantalla de 10.1" a pesar de su elevado costo, la cual ofrece una mejor resolución para que el usuario aprecie de forma clara la interfaz gráfica. Como se puede apreciar en la Figura 3.4 la pantalla es táctil y brinda la facilidad de ser manipulado por las personas que se encuentren al cuidado del paciente.

Figura 3. 4: Pantalla Touch Creen de 10.1"



Pantalla Touch Creen de 10.1", Fuente: (Banggood, s.f)

Las características de la pantalla LCD y el circuito principal son:

- Pantalla IPS con una resolución de 1280 × 800.
- LCD capacitiva, admite 10 puntos de contacto.
- Admite 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.

### 3.1.4 Parlantes

Los parlantes son usados para la reproducción de los mensajes escritos por el paciente y para la reproducción de músicas. Para ello se optó por los parlantes de la computadora como se ilustra en la Figura 3.5, ya que tiene un conector de entrada USB que permite la alimentación y la salida de tipo Plug para el audio, el cual es compatible con el conector RCA de la tarjeta RPI 3. Además, es de bajo costo y su tamaño es ideal para implementar en el prototipo. A continuación, se presenta las características más importantes.

Las características de los parlantes son:

- Conector USB para la alimentación
- Conector Plug para la salida de audio
- Control para volumen
- Bajo costo

Figura 3. 5: Parlantes para la reproducción de audio



Parlantes para la reproducción de audio, Elaborado por: Luis Collaguazo

Con el análisis hasta aquí expuesto se detallan los elementos que constituyen el hardware del prototipo.

- Raspberry PI 3
- Módulo de la cámara RPI
- Pantalla Touch de 10.1”
- Parlantes

### 3.2 Análisis de software

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

#### 3.2.1 Sistema operativo

El sistema operativo oficial de la Raspberry PI es Raspbian, sin embargo, existen otros que son compatibles los cuales se pueden utilizar de acuerdo a las necesidades del usuario. Para el desarrollo del sistema propuesto se realizó una Tabla comparativa de los diferentes sistemas operativos como se muestra en la Tabla 3.4.

Tabla 3. 4: Comparativa de sistemas operativos para la Raspberry.

<b>Sistema Operativo</b>	<b>Distribución de Linux</b>	<b>Lanzamiento Inicial</b>	<b>Interfaz gráfica por defecto</b>	<b>Compatible con Eviacam</b>
Raspbian	Debian Wheezy	Junio del 2012	Pixel	No
Kali Linux	Debian	13 Marzo 2013	GNOME	No
Ubuntu Mate	Ubuntu	Octubre 2014	Mate	Si

Comparativa de sistemas operativos para la Raspberry, Elaborado por: Luis Collaguazo

Una vez realizada la Tabla comparativa se eligió el Sistema Operativo Ubuntu Mate debido a que presenta una ventaja importante que es la compatibilidad con la aplicación de Eviacam, pues Raspbian y otros sistemas operativos no permiten la funcionalidad adecuada de dicha aplicación. A continuación, se detalla las características más importantes:

- Entorno de escritorio amigable.
- Facilidad de uso.
- Compatibles con diversas aplicaciones.
- Sistema operativo compacto.
- Distribución de Linux basado en Ubuntu.

### **3.2.2 Lenguaje de programación**

El Sistema Operativo Ubuntu Mate ofrece la posibilidad de elegir entre diferentes herramientas libres de programación como es el caso de Python, un lenguaje robusto, amigable, y fácil de utilizar. Python está disponible en dos versiones 2.7 y 3.5, para el desarrollo de la aplicación se eligió estas dos, debido a que algunas librerías disponibles en los repositorios son compatibles con la una versión y con la otra no. Python ofrece varias librerías para el desarrollo de la interfaz gráfica de usuario (GUI) como: Tkinter, WxPython, PyQt, PyGTK, la mayoría de ellas vienen preinstaladas por defecto.

Con el análisis hasta aquí expuesto se detallan todos los softwares empleados para el desarrollo del dispositivo:

- Sistema operativo: Ubuntu Mate
- Lenguaje de programación: Python
- Librerías: Tkinter, eSpeak, urllib2, json
- Aplicaciones: Eviacam, SMPlayer, Pushetta

### **3.3 Representación gráfica del algoritmo del prototipo**

El diagrama de flujo de la Figura 3.6 detalla de forma general los pasos a seguir para el desarrollo del software del prototipo de comunicación.

Figura 3. 6: Diagrama de flujo para el desarrollo del software



Diagrama de flujo para el desarrollo del software, Elaborado por: Luis Collaguazo

El diagrama de flujo de la Figura 3.7 detalla los pasos a seguir para la forma de uso del prototipo de comunicación.

Figura 3. 7: Diagrama de flujo para el uso del prototipo

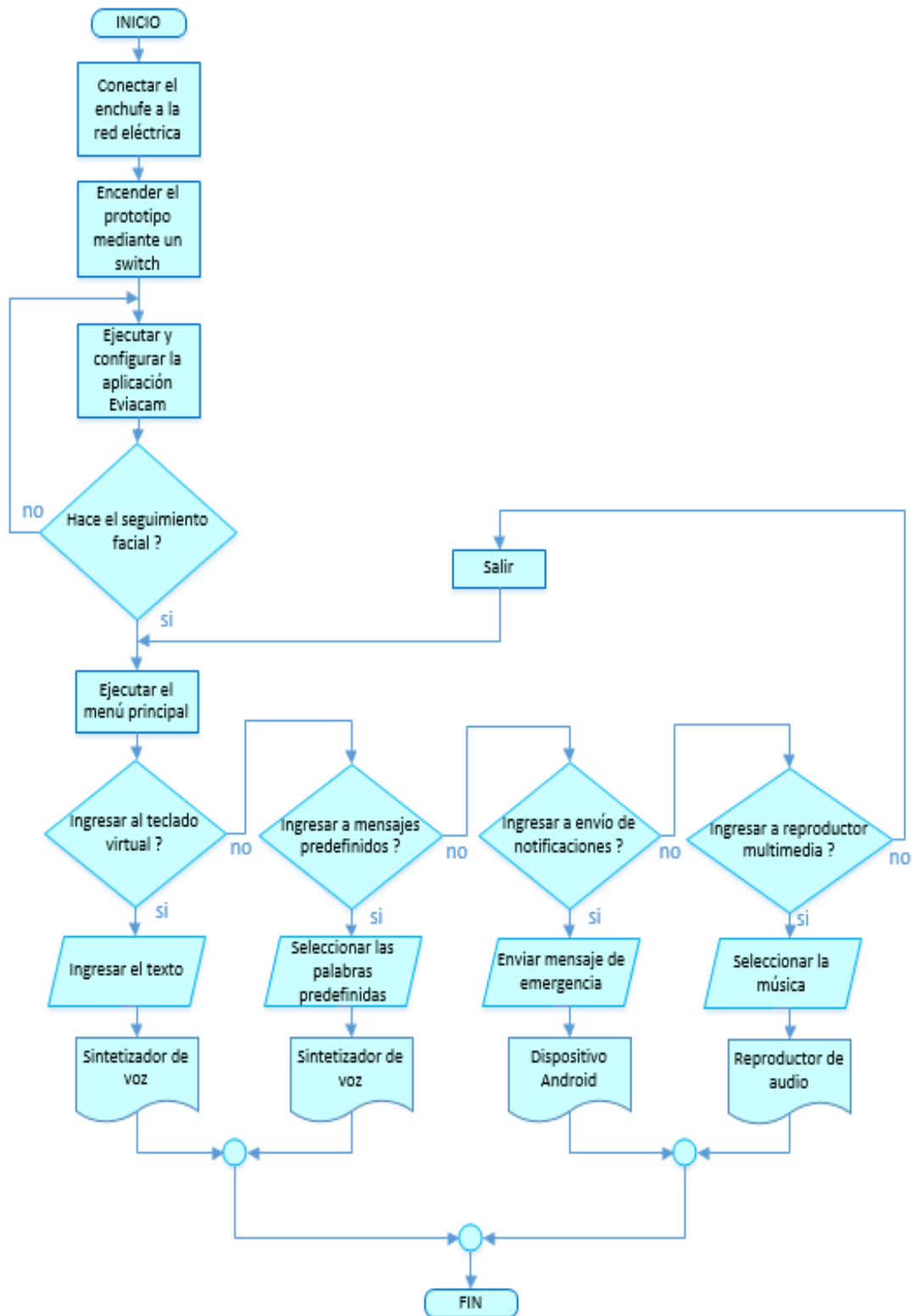


Diagrama de flujo para el uso del prototipo, Elaborado por: Luis Collaguazo



### 3.4 Implementación de hardware

El prototipo para la comunicación está diseñado con el uso de los siguientes materiales: módulo de la cámara RPI que hace el reconocimiento facial para el control del cursor del mouse, ordenador de placa reducida permite el procesamiento de los datos, pantalla Touch Screen permite visualizar la interfaz gráfica del paciente e introducir los datos a ser procesados y un altavoz para la reproducción de audio del mensaje escrito por el usuario. En la Figura 3.8 se puede apreciar el diagrama de conexión del dispositivo de comunicación.

Figura 3. 8: Diagrama esquemático

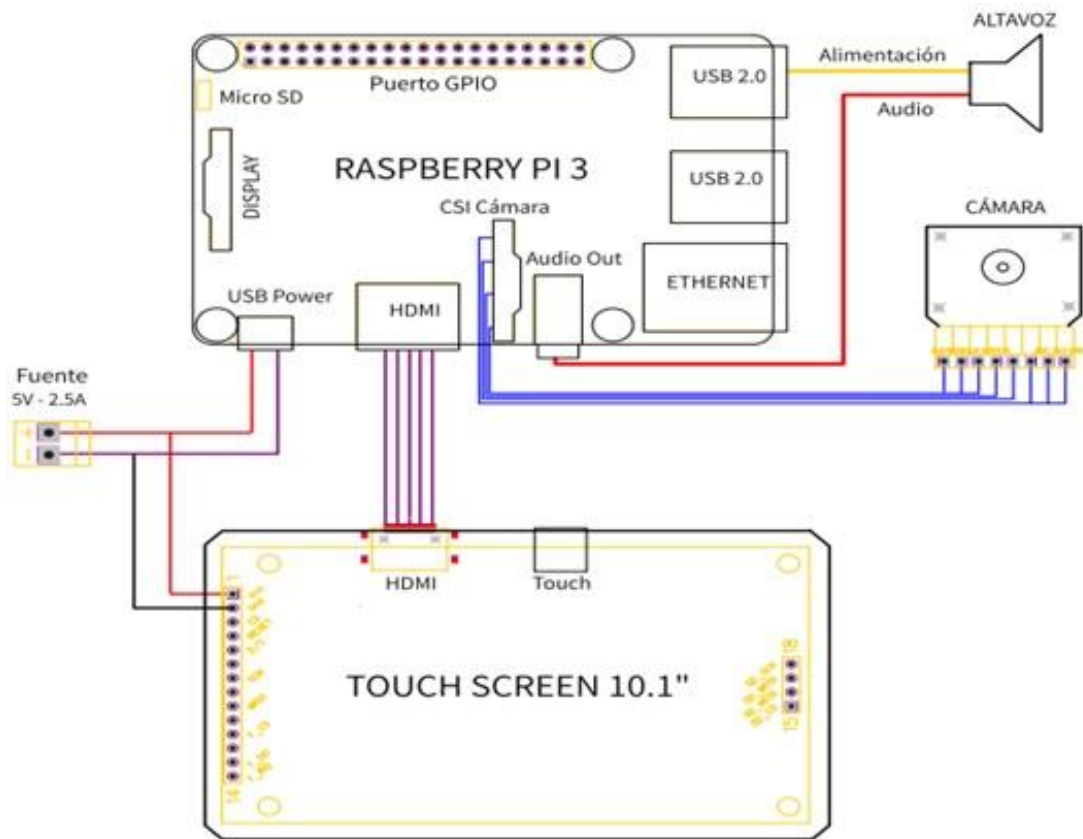


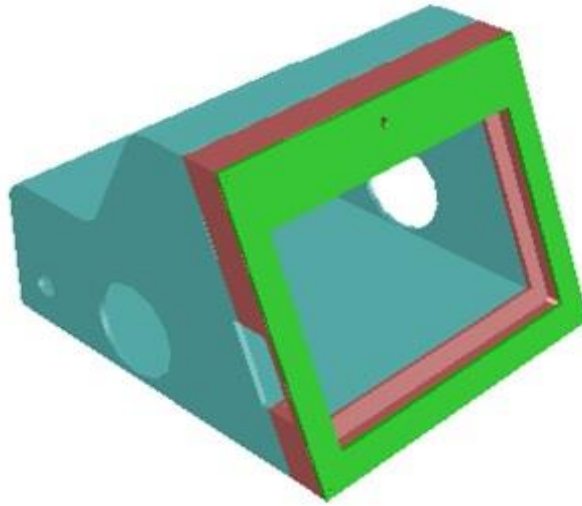
Diagrama esquemático del dispositivo de comunicación, Elaborado por: Luis Collaguazo

#### - Diseño de la estructura para el prototipo

El diseño de la estructura para el prototipo fue realizado en el software de AutoCAD, la estructura cumple la función de dar soporte y protección a cada uno de los elementos que forman el prototipo. El diseño de la estructura está realizado de acuerdo a las dimensiones y características que presenta cada elemento, el cual consta de las siguientes dimensiones: ancho 30cm, alto 24cm y profundidad 30cm. En el Anexo A

se detalla más dimensiones de la estructura. Además, en la Figura 3.9 se ilustra el diseño de la estructura elaborado en AutoCAD.

Figura 3. 9: Diseño de la estructura para el prototipo en AutoCAD



Diseño de la estructura para el prototipo, elaborado por: Luis Collaguazo

Debido al alto costo de la impresión en 3D que conlleva el presente diseño se optó por el corte a laser y el material MDF para la elaboración de la estructura. A continuación, en la Figura 3.10 se puede observar la estructura para el prototipo.

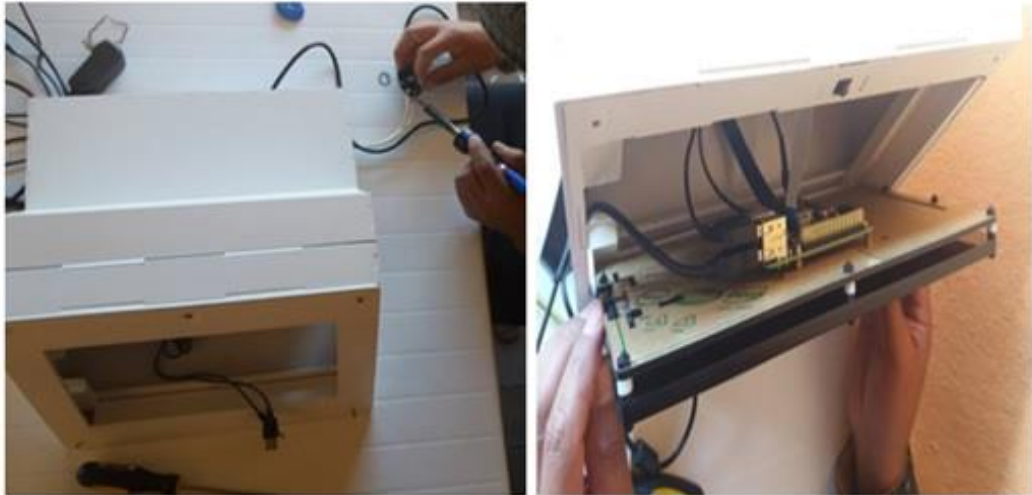
Figura 3. 10: Estructura para el prototipo



Estructura para el prototipo, Elaborado por: Luis Collaguazo

Seguidamente se procede con la conexión y ubicación de los elementos que conforman el prototipo de comunicación tal como se observa en la Figura 3.11.

Figura 3. 11: Conexión y ubicación de los elementos



Conexión y ubicación de los elementos, elaborado por: Luis Collaguazo

Finalmente, en la Figura 3.12 se observa el desarrollo del prototipo final para la comunicación de los pacientes con ELA.

Figura 3. 12: Prototipo final de comunicación

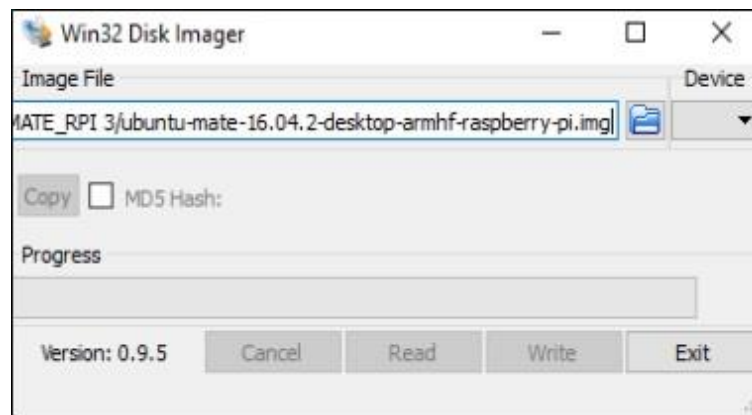


Prototipo final de comunicación, Elaborado por: Luis Collaguazo

### 3.5 Implementación de software

Se descarga el sistema operativo Ubuntu Mate desde la página oficial de la Raspberry Pi y utilizando la aplicación de win32 se carga la imagen ISO a la tarjeta micro SD de 16 GB. La micro SD debe ser de núcleo 10 para que la transferencia de datos se realice de manera óptima. En la Figura 3.13 se muestra la forma de cargar la imagen ISO a la tarjeta SD.

Figura 3. 13: Carga de Ubuntu Mate al micro SD



Carga de Ubuntu Mate al micro SD, Elaborado por: Luis Collaguazo

Seguidamente se inserta la micro SD en la placa Raspberry Pi y al momento de conectar a la fuente de alimentación se inicia la instalación del sistema operativo, durante el proceso de instalación se configura el idioma, el lugar, la fecha, el nombre del usuario y la contraseña. En la Figura 3.14 se muestra el entorno de Ubuntu Mate.

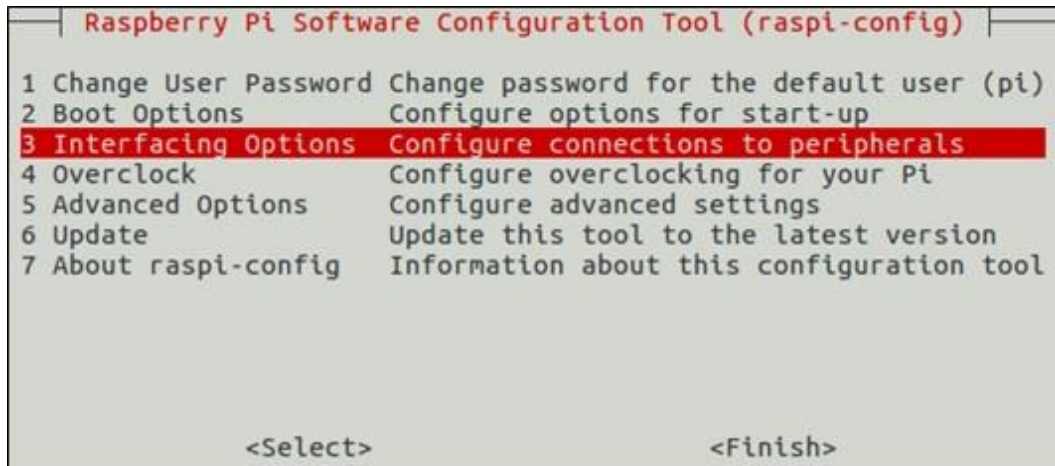
Figura 3. 14: Entorno de Ubuntu Mate



Entorno de Ubuntu Mate, Elaborado por: Luis Collaguazo

Para utilizar la cámara y otros dispositivos conectados a la placa Raspberry Pi se habilita los periféricos desde la terminal de Ubuntu Mate con el siguiente comando: `$ sudo raspi-config`. En la Figura 3.15 se muestra la ventana de configuración.

Figura 3. 15: Ventana de configuración de periféricos de RPI

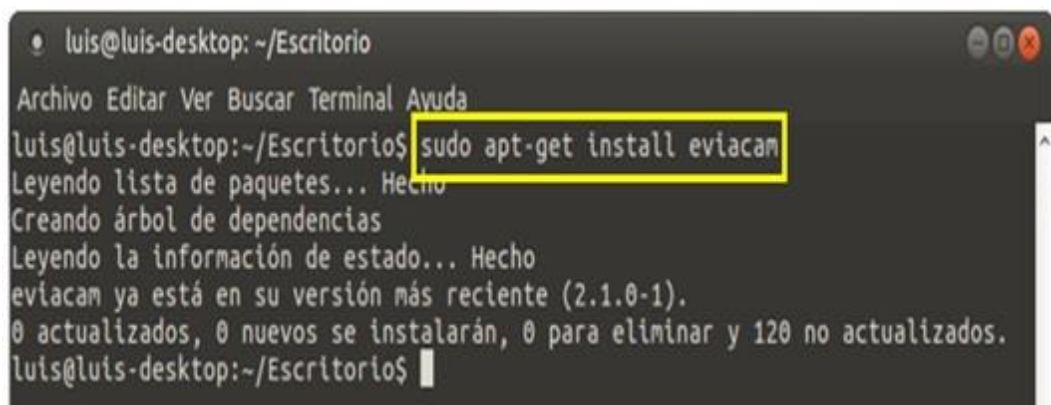


Ventana de configuración de periféricos de RPI, Elaborado por: Luis Collaguazo

### 3.5.1 Instalación de librerías y aplicaciones

Previo al desarrollo de la interfaz gráfica de comunicación se verifica que las librerías estén instaladas correctamente usando el Screen de Python, caso contrario se debe instalar utilizando el siguiente comando: `$ sudo apt-get install (nombre librería)`. De igual manera para la instalación de aplicaciones se utiliza el comando: `$ sudo apt-get install (nombre de la aplicación)`. En la Figura 3.16 se muestra la instalación de la aplicación Eviacam.

Figura 3. 16: Instalación de Eviacam



Instalación de la aplicación de Eviacam, Elaborado por: Luis Collaguazo

### 3.5.2 Desarrollo de la interfaz gráfica

Para iniciar con el desarrollo de la interfaz gráfica de usuario se importa la librería de Tkinter y se configura el constructor para la creación de la misma. En la Figura 3.17 se muestra el código empleado.

Figura 3. 17: Código empleado para la creación de la ventana

```
from Tkinter import *
import urllib2
import json
import os

#configuramos la ventana
ventanaMenu = Tk()
ventanaMenu.title("MENU")
ventanaMenu.geometry("1280x700+0+0")
ventanaMenu.config(bg = "#00FFFF")
```

Código empleado para la creación de la ventana, Elaborado por: Luis Collaguazo

El menú principal consta de 4 opciones: Simulación del Teclado Virtual, Mensajes Predefinidos, Envío de Notificaciones y Reproductor Multimedia.

**a) Simulación del Teclado Virtual.-** La programación del teclado virtual desarrollado consiste en crear botones usando el comando **Button** y a través de caracteres alfanuméricos que están agrupados en listas (**List**) se va dando forma a un teclado tradicional. La función **Lambda** permite crear una variable en una sola línea, al presionar el botón toma el dato y envía a escribir en una entrada de texto **Entry**. También se programa un diccionario de palabras, el cual radica en almacenar las palabras más empleadas del lenguaje español en un diccionario propio de Python. Esta cadena de caracteres se muestra en botones generados específicamente para referirse a diccionario de palabras que permite el ingreso de texto de forma rápida en la escritura del paciente. En la Figura 3.18 se muestra el formato para generar el diccionario de palabras en Python.

Figura 3. 18: Formato para crear el diccionario de palabras

```
dicPalabras = { "a": ["ayuda", "amigo", "aquí"]
                "b": ["bueno", "bien", "beber"]
                "c": ["cuarto", "casa", "cama"]
                }
```

Formato para crear el diccionario de palabras, Elaborado por: Luis Collaguazo

La programación de los botones adicionales del teclado virtual permite ejecutar otras acciones como es el caso del botón sintetizador de voz, que al ser pulsado convierte el texto en audio (Tecnología TTS) para ello se emplea la librería eSpeak usando el siguiente código:

```
os.system('espeak -ves "{0}"format(Cadena de texto a convertir en audio)')
```

Además, existe la forma de elegir entre la voz de hombre (-ves) o mujer (-ves+f2) y el idioma inglés (-ven) de acuerdo a las necesidades del usuario.

Del mismo modo, el botón BloqMayus permite cambiar los caracteres del teclado de minúscula a mayúscula usando el método **caracter.upper()** y de mayúsculas a minúsculas usando **caracter.lower()**.

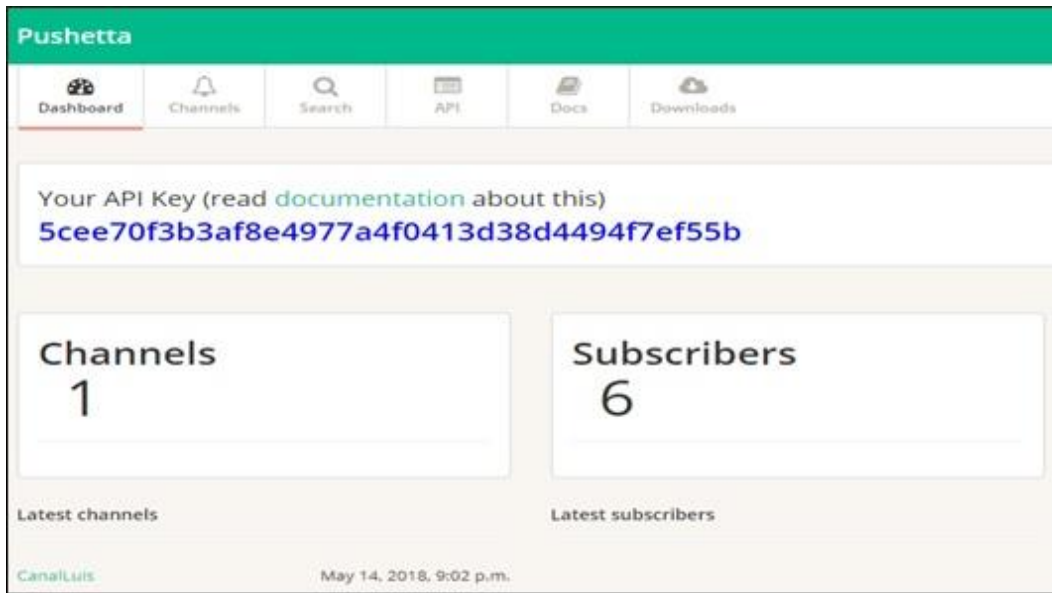
Finalmente, la simulación del Teclado Virtual ofrece otros botones con diferentes funciones como son: borrar letra, borrar todo, space y tab.

**b) Mensajes predefinidos.** - Para el desarrollo de estos mensajes se emplea el mismo método de programación descrita en el proceso del teclado virtual, con la diferencia que los botones creados contienen palabras definidas de necesidades y deseos que aprovecha el paciente para una comunicación rápida.

**c) Envío de notificaciones.** - El envío de notificaciones se realiza mediante la tecnología Push, de manera que el servidor inicia la petición al cliente para enviar un mensaje de notificación a un dispositivo Android.

El proceso que se realiza para el envío de dicha notificación es: crear una cuenta en la página oficial de pushetta.com, posteriormente se ingresa como usuario y se crea un canal para obtener el ApiKey (Token) como se puede observar en la Figura 3.19. Seguidamente se descarga e instala la aplicación de Pushetta en un teléfono Android.

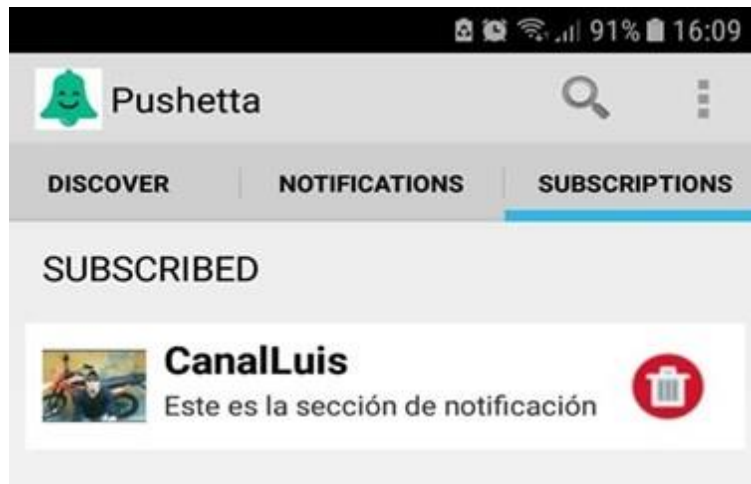
Figura 3. 19: Canal y ApiKey creado en Pushetta



Canal y ApiKey creado en Pushetta, Elaborado por: Luis Collaguazo

Después se ejecuta la aplicación y se añade el canal creado por el cual se recibirá el mensaje de notificación como se muestra en la Figura 3.20.

Figura 3. 20: Suscripción en el teléfono Android



Suscripción en el teléfono Android, Elaborado por: Luis Collaguazo

Finalmente, se implementa el código en Python importando las librerías *urllib2* y *json* los cuales permiten abrir la URL de la página Web de Pushetta y tener acceso a ella mediante API Key y el canal del usuario, seguidamente se configura el cuerpo principal de programación para la sincronización y la transmisión de datos desde el



servidor hacia el dispositivo Android. Este mensaje es ejecutado al presionar un botón desde el menú principal. A continuación, en la Figura 3.21 se muestra el código empleado para el envío del mensaje.

Figura 3. 21: Código empleado para el envío del mensaje de emergencia

```
import urllib2
import json
import os

x = "5cee70f3b3af8e4977a4f0413d38d4494f7ef55b"
y = "CanalLuis"
z = "!!! MENSAJE DE EMERGENCIA !!! (PACIENTE - ELA)"

def envioN(token, channel, message):
    dato = {"body": message, "message_type": "text/plain"}
    req = urllib2.Request('http://api.pushetta.com/api/pushes/{0}'.format(channel))
    req.add_header('Content-Type', 'application/json')
    req.add_header('Authorization', 'Token {0}'.format(token))
    respuesta= urllib2.urlopen(req, json.dumps(dato))

envioN(x,y,z)
os.system('espeak -ves "Mensaje Enviado"')
```

Código empleado para el envío del mensaje de emergencia, Elaborado por: Luis Collaguazo

**d) Reproductor multimedia.** - Python ofrece diversas funcionalidades como abrir archivos, documentos de texto, aplicaciones, entre otros. Aprovechando de esta herramienta se accede a la aplicación de SMPlayer desde un botón que se encuentra en el menú principal, para ello se utiliza el comando: *os.popen ("smplayer")*. Esta aplicación permite añadir una lista de reproducción con las músicas que se encuentran ubicadas en una carpeta externa, Además, permite configurar los iconos tales como: adelantar, retroceder, pausar y cerrar. Estas opciones se cambiaron pensando en la facilidad de uso que tenga el paciente.

El control del puntero del mouse se tiene por medio del seguimiento facial. La aplicación de Eviacam realiza este proceso a través de leves movimientos de la cabeza que detecta el módulo de la cámara RPI. De esta manera el paciente controla la interfaz gráfica detallada anteriormente.

El software (la programación) realizada se detalla en el Anexo B.

## CAPÍTULO 4

### PRUEBAS Y RESULTADOS

En el presente capítulo se detalla las pruebas realizadas del dispositivo de comunicación para verificar la funcionalidad de la interfaz gráfica y del seguimiento facial para el control del mouse, además se analizan los resultados obtenidos y se presentan las conclusiones y recomendaciones para futuros trabajos.

#### 4.1 Funcionamiento del prototipo final

El prototipo se debe conectar a una red eléctrica y para encender se activa un switch que se encuentra ubicado en la parte lateral derecha. Al terminar este proceso la pantalla muestra el escritorio de Ubuntu Mate y en ella se encuentra la carpeta *Proyecto\_Final*, como se ilustra en la Figura 4.1.

Figura 4. 1: Prototipo final de comunicación

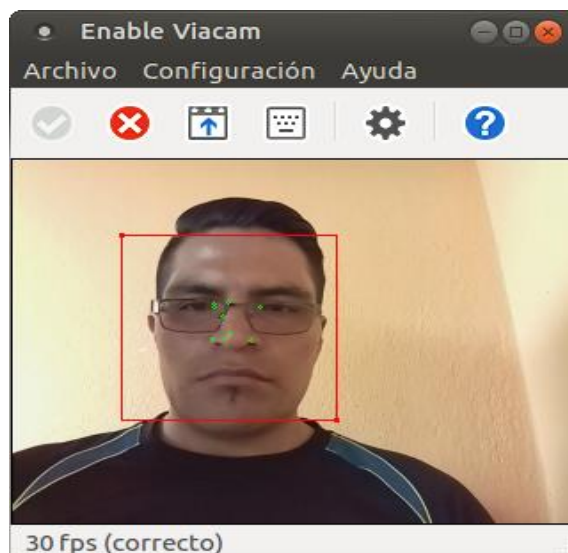


Prototipo final de comunicación, Elaborado por: Luis Collaguazo

## 4.2 Seguimiento facial

Para realizar el seguimiento facial se debe ejecutar la aplicación de Eviacam, la cual está basada en el método de visión artificial. Seguidamente se debe configurar los parámetros de tiempo, velocidad y movimiento, ajustando así a la capacidad de movimiento del paciente, como se detalla en el Anexo C. De esta manera se tiene el control del cursor del mouse para acceder a las diferentes funciones de la interfaz gráfica mediante el seguimiento facial, la simulación del clic izquierdo del mouse es cuando el paciente se detenga fijamente sobre un botón por un tiempo de 2 segundos aproximadamente. Esta prueba se enfoca para pacientes con etapa tardía de ELA, porque ellos al menos tienen movilidad de la cabeza. En la Figura 4.2 se muestra el seguimiento facial a través de Eviacam.

Figura 4. 2: Seguimiento facial



Seguimiento facial a través de la aplicación Eviacam, Elaborado por: Luis Collaguazo

En la Figura 4.2 se observa que la aplicación ejecuta satisfactoriamente y el reconocimiento se realiza de manera correcta a una imagen máxima de 30 fps (fotogramas por segundo), sin embargo en lugares donde hay mucho reflejo de luz sobre la cámara el reconocimiento disminuye a 27 fps, lo cual afecta de forma mínima el seguimiento facial para el control del puntero del mouse. El nivel de iluminación óptimo para el uso del prototipo es en un rango de 200 a 360 luxes, de acuerdo a los valores obtenidos con el medidor luxómetro.

### 4.3 Interfaz gráfica

En el escritorio de Ubuntu Mate se tiene la carpeta *Proyecto\_Final*, al acceder a ella se encuentran varios archivos y ficheros desarrollados en Python, entre ellos está un ejecutable con el nombre de *Menu.py*, que al ser ejecutado muestra la interfaz gráfica para el paciente (Menú Principal), como se ilustra en la Figura 4.3

Figura 4. 3: Menú principal para el paciente



Menú principal para el paciente, Elaborado por: Luis Collaguazo

Como se observa en la interfaz gráfica, existen 4 funcionalidades que ayudaran a la comunicación del paciente. Además, la interfaz tiene un entorno amigable y su tamaño es ideal para que el paciente se adapte con facilidad.

#### 4.3.1 Teclado virtual

Al presionar el primer botón se ejecuta la ventana del Teclado Virtual como se muestra en la Figura 4.4. Se observa que la escritura se realiza rápidamente gracias al diccionario de palabras que contiene cada una de las letras del abecedario, este mensaje escrito por el paciente es convertido en audio y es reproducido con una voz digitalizada al pulsar el botón *vozM*. Para salir de la aplicación se pulsa el botón *Atrás* regresando así al menú principal.

Figura 4. 4: Teclado virtual para el paciente

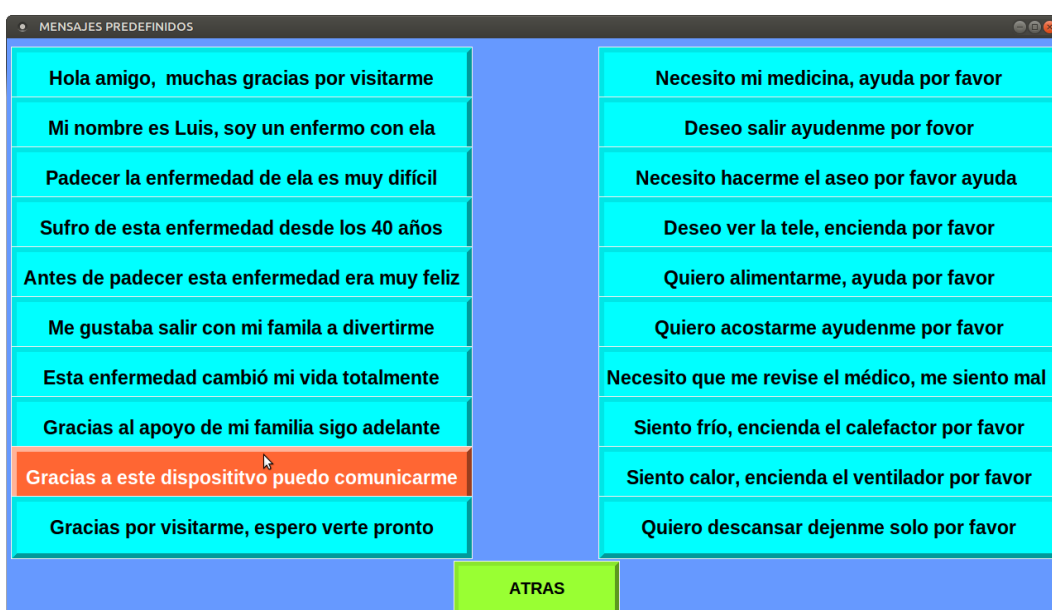


Teclado virtual para el paciente, Elaborado por: Luis Collaguazo

### 4.3.2 Mensajes predefinidos

Al presionar el segundo botón se ejecuta la ventana de Mensajes Predefinidos como se muestra en la Figura 4.5. En esta ventana se encuentran mensajes y necesidades básicas del paciente, las mismas que al ser presionadas se convierten en audio y agilitan la comunicación, los mencionados mensajes pueden ser modificados y configurados dependiendo de la forma como quiere expresar el enfermo.

Figura 4. 5: Ventana de mensajes predefinidos



Ventana de mensajes predefinidos, Elaborado por: Luis Collaguazo

### 4.3.3 Notificación Push

Al presionar el tercer botón, como se muestra en la Figura 4.6 (a), se envía el mensaje de emergencia hacia un teléfono Android para notificar que el paciente necesita ayuda. En la Figura 4.6 (b) se observa que el mensaje llega correctamente.

Figura 4. 6: Mensaje de notificación

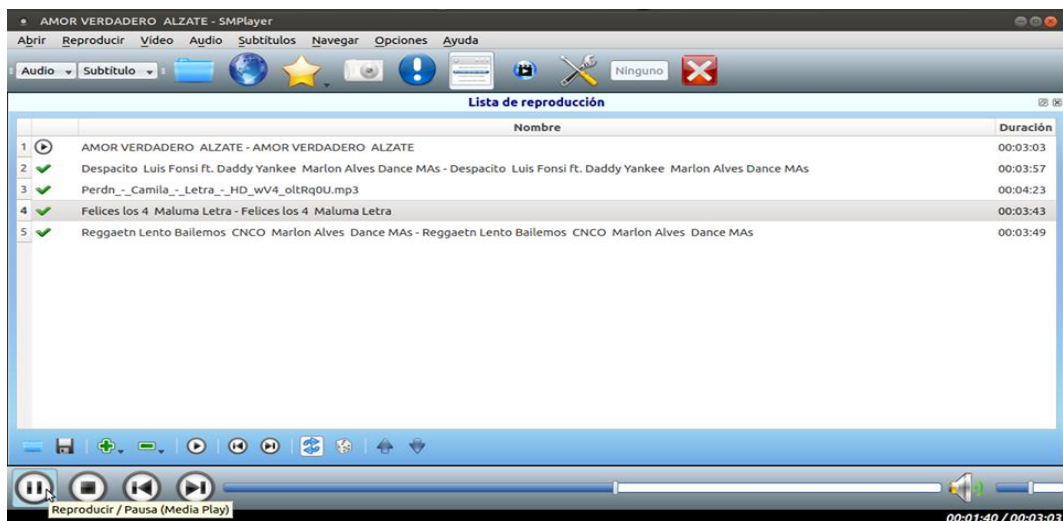


Mensaje de notificación, Elaborado por: Luis Collaguazo

### 4.3.4 Reproductor multimedia

Al presionar el cuarto botón, se ejecuta el reproductor multimedia con todas las canciones que se encuentran añadidas en la lista de reproducción como se muestra en la Figura 4.7. Esta prueba de funcionamiento resultó ser favorable debido a que el paciente se encuentra la mayor parte del tiempo solo y mediante esta opción puede escuchar música para relajarse y reducir la depresión.

Figura 4. 7: Reproductor SMPlayer



Reproductor SMPlayer, Elaborado por: Luis Collaguazo

Finalmente, el dispositivo desarrollado para la comunicación de los pacientes con ELA funcionó de manera esperada, cumpliendo así con todos los parámetros establecidos, lo cual indica que el dispositivo es de ayuda en personas con deficiencias motoras.

#### 4.4 Validación del prototipo

Las pruebas de funcionamiento se realizaron con 12 personas voluntarias quienes no sufren de ninguna discapacidad física, de distinta edad y género. Esto se realizó debido a que no se pudo encontrar una persona con la enfermedad de ELA, sobretodo en la etapa denominada tardía, para los análisis se realizó una calibración previa. Se diseñó una prueba dividido en cinco partes. La prueba consiste en medir el tiempo que cada persona toma en escribir un determinado mensaje. Este tiempo es tomado de las dos formas, lo uno escribiendo letra a letra y lo otro escribiendo con la ayuda del diccionario de palabras. Esta escritura se hace utilizando únicamente el movimiento de la cabeza.

##### - El primer mensaje escrito es: *“hola mundo”*

En la Tabla 4.1 se puede observar el tiempo empleado por cada usuario en la escritura del mensaje, aplicando la fórmula matemática se obtuvo el promedio del tiempo de la escritura letra a letra que es de 35.33 [s] y con la ayuda del diccionario de palabras que es de 18.36 [s].

Tabla 4. 1: Análisis del tiempo del primer mensaje

Prueba / Usuarios	Tiempo [s]	
	Escritura letra a letra	Escritura con diccionario
Usuario 1	34.86	17.11
Usuario 2	36.29	15.99
Usuario 3	34.12	19.67
Usuario 4	38.17	24.64
Usuario 5	33.56	19.76
Usuario 6	35.84	15.96
Usuario 7	33.62	16.23
Usuario 8	35.09	20.04
Usuario 9	34.69	15.02
Usuario 10	37.03	18.07
Usuario 11	36.29	20.59
Usuario 12	34.44	17.28
<b>Promedio</b>	<b>35.33</b>	<b>18.36</b>

Análisis del tiempo empleado en la escritura del primer mensaje, Elaborado por: Luis Collaguazo

- **El segundo mensaje escrito es: “hola amigo como estas”**

En la Tabla 4.2 se puede observar el tiempo empleado por cada usuario en la escritura del mensaje, aplicando la fórmula matemática se obtuvo el promedio del tiempo de la escritura letra a letra que es de 69.75 [s] y con la ayuda del diccionario de palabras que es de 30.44 [s].

Tabla 4. 2: Análisis del tiempo del segundo mensaje

<b>Prueba / Usuarios</b>	<b>Tiempo [s] Escritura letra a letra</b>	<b>Tiempo [s] Escritura con diccionario</b>
Usuario 1	68.03	30.30
Usuario 2	68.19	28.37
Usuario 3	67.96	29.88
Usuario 4	66.71	29.53
Usuario 5	71.89	34.17
Usuario 6	68.54	29.78
Usuario 7	68.75	30.12
Usuario 8	68.32	31.15
Usuario 9	69.81	29.92
Usuario 10	69.22	30.45
Usuario 11	69.14	31.03
Usuario 12	70.02	30.56
<b>Promedio</b>	<b>69.75</b>	<b>30.44</b>

Análisis del tiempo empleado en la escritura del segundo mensaje, Elaborado por Luis: Collaguazo



- **El tercer mensaje escrito es: “Padecer la enfermedad de ELA es muy difícil”**

En la Tabla 4.3 se puede observar el tiempo empleado por cada usuario en la escritura del mensaje, aplicando la fórmula matemática se obtuvo el promedio del tiempo de la escritura letra a letra que es de 115.81 [s] y con la ayuda del diccionario de palabras que es de 59.42 [s].

Tabla 4. 3: Análisis del tiempo del tercer mensaje

<b>Prueba / Usuarios</b>	<b>Tiempo [s] Escritura letra a letra</b>	<b>Tiempo [s] Escritura con diccionario</b>
Usuario 1	116.91	63.87
Usuario 2	113.93	56.50
Usuario 3	114.54	56.13
Usuario 4	116.17	58.88
Usuario 5	112.39	55.69
Usuario 6	115.36	56.39
Usuario 7	117.69	62.14
Usuario 8	117.33	59.31
Usuario 9	115.09	57.67
Usuario 10	118.03	65.07
Usuario 11	116.27	62.45
Usuario 12	115.98	58.96
<b>Promedio</b>	<b>115.81</b>	<b>59.42</b>

Análisis del tiempo empleado en la escritura del tercer mensaje, Elaborado por: Luis Collaguazo

- **El cuarto mensaje escrito es: “*gracias al apoyo y el esfuerzo de mi familia sigo adelante*”**

En la Tabla 4.4 se puede observar el tiempo empleado por cada usuario en la escritura del mensaje, aplicando la fórmula matemática se obtuvo el promedio del tiempo de la escritura letra a letra que es de 168.31 [s] y con la ayuda del diccionario de palabras que es de 83.95 [s].

Tabla 4. 4: Análisis del tiempo del cuarto mensaje

<b>Prueba / Usuarios</b>	<b>Tiempo [s] Escritura letra a letra</b>	<b>Tiempo [s] Escritura con diccionario</b>
Usuario 1	172.47	87.58
Usuario 2	166.51	81.14
Usuario 3	166.94	83.35
Usuario 4	167.78	83.58
Usuario 5	167.62	82.04
Usuario 6	169.18	85.67
Usuario 7	167.36	81.89
Usuario 8	169.24	87.08
Usuario 9	166.93	82.44
Usuario 10	168.07	84.32
Usuario 11	170.68	86.04
Usuario 12	166.89	82.29
<b>Promedio</b>	<b>168.31</b>	<b>83.95</b>

Análisis del tiempo empleado en la escritura del cuarto mensaje, Elaborado por: Luis Collaguazo

- **El quinto mensaje escrito es: “*gracias a este dispositivo me puedo comunicar con las personas que me rodean*”**

En la Tabla 4.5 se puede observar el tiempo empleado por cada usuario en la escritura del mensaje, aplicando la fórmula matemática se obtuvo el promedio del tiempo de la escritura letra a letra que es de 213.07 [s] y con la ayuda del diccionario de palabras que es de 103.42 [s].

Tabla 4. 5: Análisis del tiempo del quinto mensaje

<b>Prueba / Usuarios</b>	<b>Tiempo [s] Escritura letra a letra</b>	<b>Tiempo [s] Escritura con diccionario</b>
Usuario 1	212.78	101.55
Usuario 2	216.34	115.38
Usuario 3	213.32	103.82
Usuario 4	211.76	99.09
Usuario 5	213.31	108.14
Usuario 6	209.52	96.56
Usuario 7	215.59	109.37
Usuario 8	214.16	98.32
Usuario 9	211.02	97.63
Usuario 10	213.04	105.33
Usuario 11	212.88	106.06
Usuario 12	213.14	99.77
<b>Promedio</b>	<b>213.07</b>	<b>103.42</b>

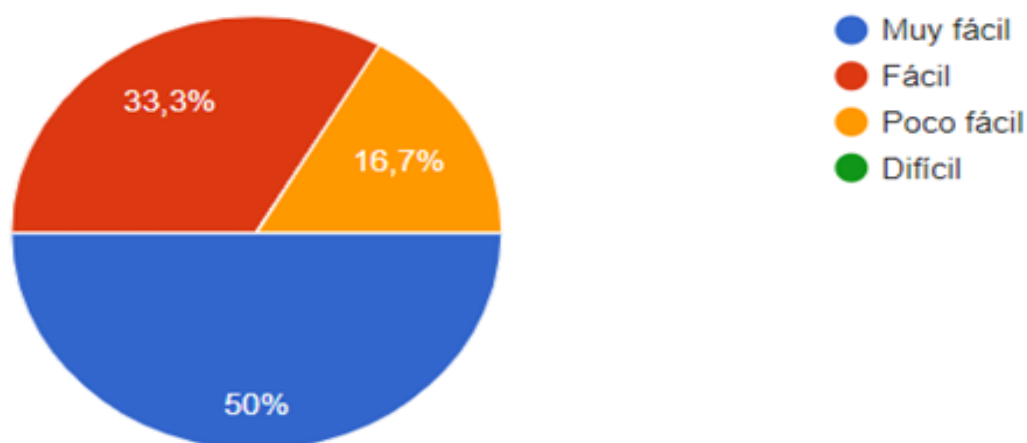
Análisis del tiempo empleado en la escritura del quinto mensaje, Elaborado por: Luis Collaguazo

Además de realizar el análisis del tiempo empleado en la escritura de los diferentes mensajes, se realizó la metodología de las encuestas para saber la aceptación que tiene el prototipo de comunicación. Para esto se tomó como muestra a 12 personas quienes después de utilizar el sistema tuvieron la apertura de responder las preguntas planteadas.

El resultado de la encuesta aplicada, arrojó los siguientes datos:

**Pregunta 1: ¿Qué tan fácil fue usar el prototipo de comunicación?**

Figura 4. 8: Representación gráfica del resultado de la pregunta 1

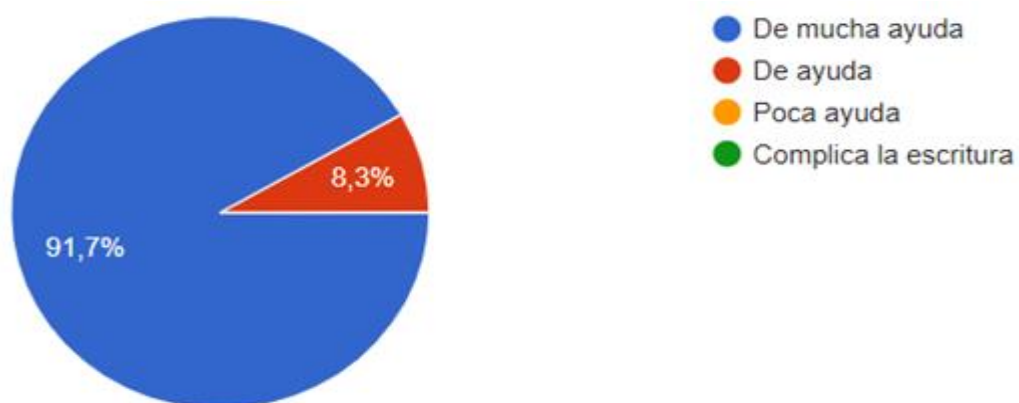


Gráfica del resultado sobre del uso del prototipo, Elaborado por: Luis Collaguazo.

En la Figura 4.8 se muestra que el 50% de las personas encuestadas les resultó muy fácil el utilizar el prototipo de comunicación, esto significa que en cuanto al manejo del prototipo no se deben realizar cambios en el funcionamiento del dispositivo

**Pregunta 2: ¿Considera usted al diccionario de palabras que contiene el teclado virtual, como una ayuda en la escritura rápida del mensaje?**

Figura 4. 9: Representación gráfica del resultado de la pregunta 2

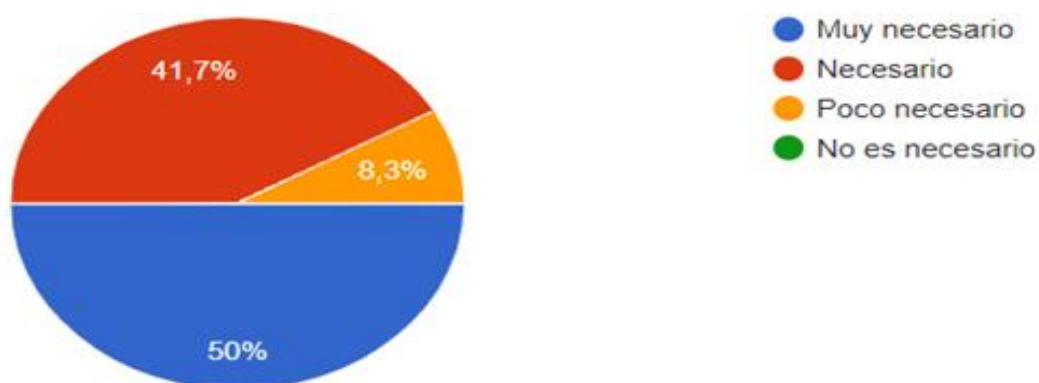


Gráfica del resultado de la importancia del diccionario de palabras en la escritura del mensaje, Elaborado por: Luis Collaguazo.

En la Figura 4.9 se muestra que el 91,7% de las personas encuestadas les resultó de mucha ayuda el uso del diccionario de palabras tanto para la escritura rápida del mensaje disminuyendo el tiempo y cómodo para la facilidad de comunicación.

**Pregunta 3: ¿Le parece necesario la opción de enviar el mensaje de notificación hacia el dispositivo Android de un familiar?**

Figura 4. 10 Representación gráfica del resultado de la pregunta 3

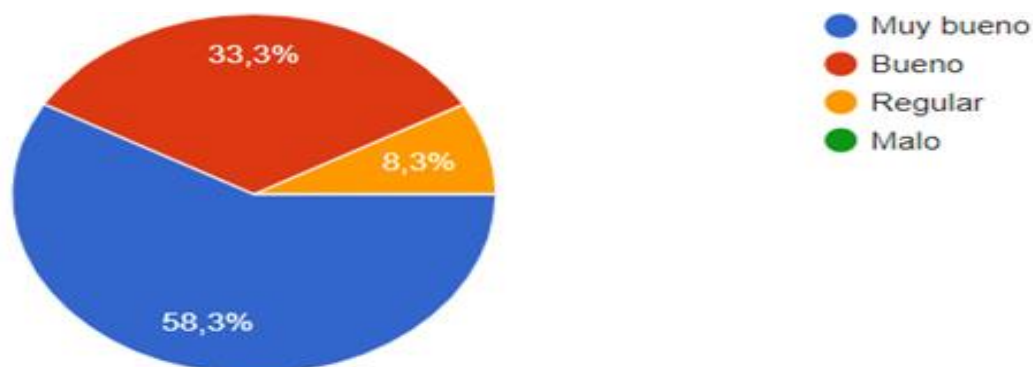


Gráfica del resultado a cerca del envío de mensajes de notificaciones, Elaborado por: Luis Collaguazo

En cuanto al envío de notificaciones, en la Figura 4.10 se muestra que el 50% de las personas encuestadas manifiestan que es muy necesaria la opción de envío de notificación que tiene el dispositivo, lo cual permite alertar sobre alguna emergencia a sus familiares y amigos.

**Pregunta 4. ¿Cómo le pareció el seguimiento facial para el control del cursor?**

Figura 4. 11: Representación gráfica del resultado de la pregunta 4

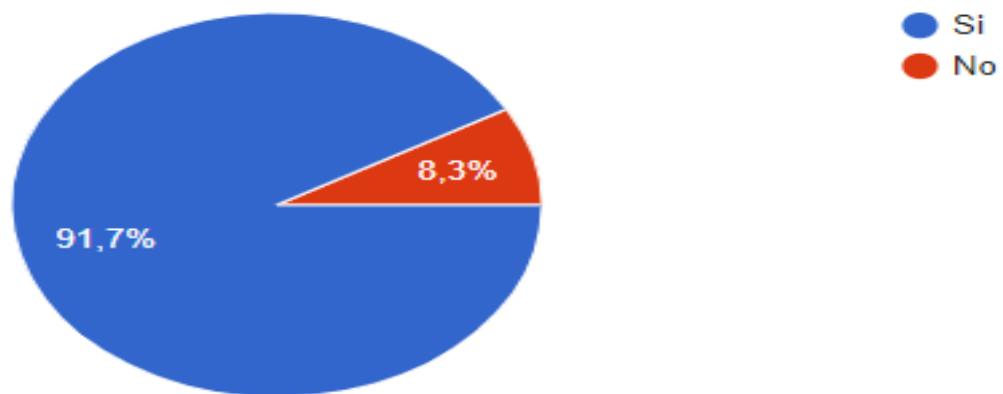


Gráfica del resultado a cerca del seguimiento facial, Elaborado por: Luis Collaguazo

En la Figura 4.11 se muestra que el 58.3% de las personas encuestadas manifiestan que el seguimiento facial que hace la aplicación de Eviacam es muy bueno e interesante debido a que la aplicación ejecuta satisfactoriamente y el reconocimiento se realiza de manera esperada a una imagen de 30 fps (fotogramas por segundo) lo cual indica que es correcto para el control del puntero del mouse.

**Pregunta 5. ¿Después de usar el prototipo, usted desearía adquirir el dispositivo?**

Figura 4. 12: Representación gráfica del resultado de la pregunta 5



Gráfica del resultado si el prototipo pudiera ser rentable en el mercado, Elaborado por: Luis Collaguazo

En la Figura 4.12 se muestra que el 91.7% de las personas encuestadas manifiestan que si desean adquirir el prototipo de comunicación, debido a que en el país no existen muchos dispositivos que faciliten la comunicación de las personas que sufren de la enfermedad de ELA.

**4.5 Análisis de resultados**

De las pruebas realizadas anteriormente en la escritura de mensajes se obtuvo el promedio del tiempo en la escritura letra a letra y en la escritura haciendo uso del diccionario de palabras, con estos datos y aplicando la ecuación (1) se obtiene el tiempo por palabras y a su vez el promedio como se observa en la Tabla 4.6.

$$Tiempo\ por\ palabras = \frac{Tiempo\ de\ la\ escritura\ del\ mensaje}{Número\ de\ palabras\ del\ mensaje} \quad Ec.\ (1)$$

Tabla 4. 6: Promedio del tiempo por palabra

Mensajes	Número de palabras	Tiempo del mensaje escritura letra a letra [s]	Tiempo del mensaje escritura con diccionario [s]	Tiempo por palabra escritura letra a letra [s]	Tiempo por palabra escritura con diccionario [s]
1	2	35.33	18.36	17.67	9.18
2	4	69.75	30.44	17.44	7.61
3	8	115.81	59.42	14.48	7.43
4	11	168.31	83.95	15.30	7.63
5	13	213.07	103.42	16.39	7.96
<b>Promedio por palabra</b>				<b>16.26</b>	<b>7.96</b>

Promedio del tiempo por palabra, Elaborado por: Luis Collaguazo

Como se observa en la Tabla 4.6 el tiempo empleado en la escritura de una palabra haciendo uso del diccionario de palabras reduce en 8.3 [s] con respecto a la escritura letra a letra.

#### 4.6 Análisis económico

##### 4.6.1 Costo de hardware

A continuación, en la Tabla 4.7 se detalla el costo de los dispositivos y los componentes que fueron utilizados en el desarrollo del prototipo para la comunicación de las pacientes con ELA.

Tabla 4. 7: Costo de hardware

Elemento	Cantidad	Valor unitario (USD)	Valor total (USD)
Raspberry Pi3	1	35	35
Cámara RPI	1	28	28
Micro SD 16GB	1	10	10
Pantalla Touch Screen de 10.1"	1	132	132
Parlantes de PC Hv-sk430	1	8.50	8.50
Cargador de 5V a 2.5 A	2	10	20
Interruptor ON/OFF con Luz	1	0.70	0.70
Cable de Alimentación con Enchufe + socket JR-101	1	4.80	4.80
Maqueta del prototipo material MDF	1	50	50
Materiales Adicionales	-	-	5
<b>Total</b>			<b>280</b>

Costo de dispositivos y componentes para el desarrollo del prototipo, Elaborado por: Luis Collaguazo

#### 4.6.2 Costo de diseño

Para la valoración del costo del diseño se tomó en cuenta el total de horas trabajadas en el proyecto. La remuneración mensual equivalente a un Ingeniero en Electrónica de acuerdo a la resolución N°. MRL 2012-0021 establecido por el Ministerio de Relaciones Laborales es de \$ 817 que corresponde a la escala de un Servidor Público1. (Ministerio de Relaciones Laborales, 2012)

- Mensual: \$ 817
- Diario:  $\$ 817 / 21 = \$ 38.90$
- Hora:  $\$ 38.90 / 8 = \$ 4.86$

Se toma como tiempo estimado de trabajo un total de 120 horas de investigación, distribuidas en la realización del diseño, la programación y las pruebas de funcionamiento.

Costo de diseño = Número de horas trabajadas \* valor de hora

Costo de diseño =  $120 * \$ 4.86$

**Costo de diseño = \$ 583.2**

#### 4.6.3 Costo total del prototipo

En la Tabla 4.8 se presenta el análisis del costo total del prototipo sumado los costos de hardware y diseño, el cual es desarrollado para facilitar la comunicación de los pacientes con ELA.

Tabla 4. 8: Costo total del prototipo

Descripción	Valor (USD)
Costo del hardware	280
Costo del diseño	583.2
<b>COSTO TOTAL</b>	<b>863.2</b>

Costo total de la elaboración del prototipo, Elaborado por: Luis Collaguazo

Después de haber obtenido el costo final del prototipo, se procede a fijar el precio de venta del prototipo aplicando la ecuación (2).



$$\text{Precio de venta} = \frac{\text{Costo del prototipo}}{100 - \text{porcentaje de beneficio}} \times 100 \quad \text{Ec. (2)}$$

$$\text{Precio de venta} = \frac{\$ 863.2}{100 - 50} \times 100$$

$$\text{Precio de venta} = \frac{\$ 863.2}{50} \times 100$$

$$\text{Precio de venta} = \$ 1726.4$$

El valor total para implementar el prototipo para la comunicación de pacientes con ELA es de \$ 1726.4. Este valor es bajo en comparación con el valor de los dispositivos comerciales que varían de \$ 4000 a \$ 15000 aproximadamente, en el Anexo D se presenta la cotización de estos dispositivos. En la Tabla 4.9 se muestra el estudio comparativo entre el prototipo desarrollado y el dispositivo comercial EyeMobile Mini de la empresa Tobii Dynavox desde el criterio del paciente que adquiere el equipo.

Tabla 4. 9: Comparativa del prototipo desarrollado y el dispositivo EyeMobile Mini

Detalles	Dispositivo desarrollado	EyeMobile Mini
Imagen del dispositivo		
Disponibilidad	Open Source	Protegido
Control cursor	Seguimiento facial	Rastreador Ocular
Pantalla	Táctil	Táctil
Funciones	<ul style="list-style-type: none"> <li>- Teclado virtual</li> <li>- Diccionario de palabras</li> <li>- Sintetizador de voz</li> <li>- Reproductor de música</li> <li>- Envío de notificaciones</li> </ul>	<ul style="list-style-type: none"> <li>- Interfaz para el acceso a internet, sitios de redes sociales y juegos</li> <li>- Permite Reproducir músicas y ver películas</li> <li>- Permite escribir texto</li> </ul>
Precio	\$ 1726.4	\$ 4759.00

Comparativa entre el prototipo desarrollado y el dispositivo comercial EyeMobile Mini, Elaborado por: Luis Collaguazo

Para realizar la relación costo beneficio se tomó en consideración el precio total del dispositivo comercial y el precio total del prototipo desarrollado, los cuales ofrecen las mismas funcionalidades de ayuda para la comunicación de los pacientes con ELA. Dicha relación se denota en la ecuación (3). (Barros & López, 2016)

$$\frac{B}{C} = \frac{\text{Costo dispositivo comercial}}{\text{Costo prototipo desarrollado}} \quad \text{Ec. (3)}$$

$$\frac{B}{C} = \frac{4759.00}{1726.4}$$

$$\frac{B}{C} = 2.76$$

El indicador define que el proyecto es rentable cuando la relación costo beneficio es mayor que uno, por lo tanto, como se obtuvo un valor resultante mayor que uno se considera que el dispositivo es exitoso. Además se puede obtener el porcentaje de ahorro en base a la ecuación (4). (Barros & López, 2016)

$$\%Ahorro = \frac{\text{costo inicial} - \text{costo final}}{\text{costo inicial}} \times 100\% \quad \text{E. (4)}$$

$$\%Ahorro = \frac{4759.00 - 1726.4}{4759.00} \times 100\%$$

$$\%Ahorro = 63.72\%$$

En la Tabla 4.10 se muestra el estudio comparativo entre el prototipo desarrollado y el dispositivo comercial Tobii I15+ Plus de la empresa Tobii Dynavox.

Tabla 4. 10: Comparativa del prototipo desarrollado y el dispositivo Tobii I15+ Plus

Detalles	Dispositivo desarrollado	Tobii I15+ Plus
Imagen del dispositivo		
Disponibilidad	Open Source	Protegido
Control cursor	Seguimiento facial	Rastreador Ocular
Pantalla	Táctil	Táctil
Funciones	<ul style="list-style-type: none"> <li>- Teclado virtual</li> <li>- Diccionario de palabras</li> <li>- Sintetizador de voz</li> <li>- Reproductor de música</li> <li>- Envío de notificaciones</li> </ul>	<ul style="list-style-type: none"> <li>- Teclado virtual</li> <li>- Sintetizador de voz</li> <li>- Predictor de palabras</li> <li>- Funciones rápidas</li> <li>- Botón de pánico para emitir una alarma</li> </ul>
Precio	\$ 1726.4	\$ 14996.00

Comparativa entre el prototipo desarrollado y el dispositivo comercial Tobii I15+ Plus, Elaborado por: Luis Collaguazo

Se determinó la relación costo beneficio aplicando la ecuación (3) como se detalla anteriormente.

$$\frac{B}{C} = \frac{14996.00}{1726.4}$$

$$\frac{B}{C} = 8.69$$

El valor resultante es mayor que uno. En consecuencia el dispositivo es exitoso. También se obtiene el porcentaje de ahorro aplicando la ecuación (4).

$$\%Ahorro = \frac{14996.00 - 1726.4}{14996.00} \times 100\%$$

$$\%Ahorro = 88.49\%$$

Con los resultados obtenidos del análisis económico, se constata que el prototipo de comunicación resultó totalmente beneficioso debido a que el paciente ahorra el dinero en la adquisición del dispositivo.

## **4.7 Conclusiones y Recomendaciones**

### **4.7.1 Conclusiones**

Los enfermos que se encuentren en la etapa tardía pueden comunicarse mediante expresiones faciales y con el movimiento de la cabeza, razón por la cual se utilizó el módulo de la cámara RPI y la aplicación de Eviacam que permiten hacer el seguimiento facial y controlar el puntero del mouse con la cabeza. Además, la cámara se ubicó en la parte superior central de la estructura, debido a que es el lugar ideal para realizar la correcta captura del rostro en el usuario.

Se diseñó el prototipo para la comunicación del paciente con ELA utilizando la tarjeta Raspberry Pi, esta tarjeta RPI siendo de un tamaño reducido y de bajo costo cumple con las mismas funcionalidades de un ordenador normal y el lenguaje de programación utilizado es Python, las mismas que son de software y hardware libre (Open Source), que brinda la posibilidad de añadir a futuro más necesidades requeridas por el paciente sin modificar en su totalidad el código actual.

Se desarrolló la interfaz gráfica para el paciente con ELA que permite acceder a la simulación del teclado virtual, y en él escribir mensajes que son convertidos en audio mediante el sintetizador de voz (Tecnología TTS), además ofrece la posibilidad de acceder a las diferentes funciones como mensajes predefinidos, envío de notificaciones y reproductor de música, lo cual lo convierte en un sistema de comunicación alternativo y novedoso.

Se determinó que en la escritura de los mensajes haciendo uso del diccionario de palabras, reduce el promedio del tiempo en 8.3 [s] en la escritura del mensaje por cada palabra con respecto a la escritura letra a letra.

La relación costo beneficio tomando en consideración el precio total del prototipo desarrollado, con el precio total del dispositivo comercial EyeMobile Mini de la empresa Tobii Dinavox, muestra que el proyecto es rentable porque se obtuvo un valor resultante de 2.76 que es un valor mayor que 1.

La relación costo beneficio tomando en consideración el precio total del prototipo desarrollado, con el precio total del dispositivo comercial Tobii I15+ Plus de la empresa Tobii Dinavox, muestra que el proyecto es rentable porque se obtuvo un valor resultante de 8.69 que es un valor mayor que 1.

#### **4.7.2 Recomendaciones**

Utilizar el micro SD con capacidad de almacenamiento igual o superior a 16 GB y con una velocidad de transmisión de núcleo 10, para aprovechar todos los recursos del software con mayor rapidez, porque de lo contrario la tarjeta RPI 3 se tornaría lenta y tediosa para el uso.

Instalar el sistema operativo Ubuntu Mate en la tarjeta RPI 3, debido a que este software es compatible con la aplicación de Eviacam, con la cual se hace el seguimiento facial para el control del cursor del mouse.

Programar mediante la creación de métodos, brinda facilidades para utilizar en cualquier parte del programa, probar de manera independiente, hacer las modificaciones y encontrar de forma fácil el origen del error.

Calibrar la velocidad del cursor y el tiempo de espera del clic de la aplicación Eviacam, de acuerdo a la capacidad física de movimiento que tenga cada persona para el control del cursor del mouse.

## REFERENCIAS BIBLIOGRÁFICAS

- ABC TECNOLOGÍA. (24 de Junio de 2014). *Texto predictivo*. Recuperado el 05 de Octubre de 2018, de <https://www.abc.es/tecnologia/informatica-software/20140624/abci-teclado-predictivo-SwiftKey-qwerty-smartphones-telefonos-moviles-mensajes-texto-201406231516.html>
- ADELA. (Diciembre de 2014). *La enfermedad*. Recuperado el 02 de octubre de 2018, de Asociación Española de ELA: <https://adelaweb.org/la-ela/la-enfermedad/>
- ALS ASSOCIATION. (2018). *What is the ALS*. Recuperado el 02 de Octubre de 2018, de <http://www.alsa.org/en-espanol/qu-es-la-ela.html>
- Banggood. (s.f). *Pantalla Touch Screen*. Recuperado el 05 de Octubre de 2018, de [https://www.banggood.com/es/101-Inch-Capacitive-HDMI-LCD-IPS-Touch-Screen-1280x800-With-Stander-For-Raspberry-Pi-Banana-Pi-p-1069732.html?rmmds=search&cur\\_warehouse=CN](https://www.banggood.com/es/101-Inch-Capacitive-HDMI-LCD-IPS-Touch-Screen-1280x800-With-Stander-For-Raspberry-Pi-Banana-Pi-p-1069732.html?rmmds=search&cur_warehouse=CN)
- Barros, C., & López, P. (2016). Diseño y construcción de un prototipo de prótesis de pierna para amputación transfemoral. *Repositorio Institucional de la Universidad de las Fuerzas Armadas ESPE*, 1-191. Recuperado el 05 de Octubre de 2018, de <http://repositorio.espe.edu.ec/handle/21000/12095>
- Bossa, L., Abarca, M. E., Torres, E. S., Ramírez, G., & Garcia, M. (s.f). *Esclerosis Lateral Amiotrófica*. Obtenido de Manual para los pacientes con ELA.
- Cáceres, L., & Moyano, J. (s.f). *Push Technology*. Recuperado el 05 de Octubre de 2018
- Caiza, G. J., & García, M. V. (2017). Implementación de sistemas distribuidos de bajo costo bajo norma IEC-61499, en la estación de clasificación y manipulación del MPS 500. *INGENIUS Revista de Ciencia y Tecnología*(18), 40-46. Obtenido de <https://doi.org/10.17163/ings.n18.2017.05>
- Consejos de tu farmacéutico. (20 de junio de 2016). *El 21 de junio día Internacional de la Esclerosis Lateral Amiotrófica(ELA)*. Recuperado el 02 de Octubre de 2018, de <http://www.consejosdetufarmacaceutico.com/ela-espana/>
- Departamento de Tecnología. (s.f). *Sistemas Electrónicos*. Recuperado el 02 de Octubre de 2018
- ELA Andalucía. (04 de Mayo de 2017). *Irisbond tradicional*. Recuperado el 02 de Octubre de 2018, de <http://www.elaandalucia.es/WP/irisbond-tradicional-nuevo-irisbond-duo/>

- Free Software Foundation. (12 de Junio de 2018). *Free Software*. Recuperado el 05 de Octubre de 2018, de <https://www.gnu.org/philosophy/free-sw.es.html>
- gizmoj. (2015). *Raspberry Pi Camera Module v1.3*. Recuperado el 05 de Octubre de 2018, de <https://www.gizmojo.com.ar/products/raspberry-pi-camera-module-v1-3>
- Grey, D., & Cook, A. (s.f). *Tecnología de Asistencia*. Recuperado el 05 de Octubre de 2018, de <https://www.britannica.com/science/assistive-technology>
- Hernandez, J. (01 de Marzo de 2015). *Dispositivos Mixtos*. Recuperado el 05 de Octubre de 2018
- Illapa, B. (Marzo de 2017). Proyecto de inclusión digital empleando herramientas de las NTIC`S de software libre en el GADMCG, para mejorar las competencias laborales. *Repositorio Institucional de la Escuela Superior Politécnica de Chimborazo*, 1-176. doi:UDCTIPEC;20T00817
- Leira, R., & Gómez, J. (05 de Junio de 2016). *Electrónica Analógica*. Recuperado el 02 de Octubre de 2018
- Llorens, C. (29 de Diciembre de 2015). *software para manejar el ordenador con la vista*. Recuperado el 02 de Octubre de 2018, de [http://cadenaser.com/emisora/2015/12/29/radio\\_san\\_sebastian/1451398460\\_834660.html](http://cadenaser.com/emisora/2015/12/29/radio_san_sebastian/1451398460_834660.html)
- Manumateos. (02 de Septiembre de 2013). *Tecnología Push*. Recuperado el 02 de Octubre de 2018, de <https://www.genbeta.com/a-fondo/tecnologia-push-asi-funciona>
- MAYO CLINIC. (20 de Mayo de 2017). *Esclerosis Lateral Amiotrófica*. Recuperado el 02 de Octubre de 2018, de Mayo-Clinic: <https://www.mayoclinic.org/es-es/diseases-conditions/amyotrophic-lateral-sclerosis/symptoms-causes/syc-20354022>
- Ministerio de Educación y Ciencia. (s.f). *Educación*. Recuperado el 05 de Octubre de 2018, de <http://ares.cnice.mec.es/informes/17/contenido/36.htm>
- Ministerio de Relaciones Laborales. (2012). Recuperado el 05 de Octubre de 2018
- Mosquera, D. (09 de marzo de 2016). *Redacción médica*. Recuperado el 01 de octubre de 2018, de Salud pública: <https://www.redaccionmedica.ec/secciones/salud-publica/50-casos-de-ela-registrados-en-el-ecuador-87334>
- Municipalidad de San Cayetano. (2018). *Elementos electronicos de entrada, salida y proceso*. Recuperado el 02 de Octubre de 2018, de

<http://sancayetano.magnaplus.org/articulo/-/articulo/RT374/elementos-electronicos-de-entrada-salida-y-proceso>

- National Institutes of Health. (03 de Enero de 2018). *Esclerosis Lateral Amiotrofica*. Recuperado el 02 de Octubre de 2018, de [https://espanol.ninds.nih.gov/trastornos/esclerosis\\_lateral\\_amiotrofica.htm](https://espanol.ninds.nih.gov/trastornos/esclerosis_lateral_amiotrofica.htm)
- Organización Mundial de la Salud. (18 de Mayo de 2018). *Tecnología de asistencia*. Recuperado el 05 de Octubre de 2018, de <http://www.who.int/es/news-room/fact-sheets/detail/assistive-technology>
- Origen Instruments. (s.f). *Sip and Puff Switch Solutions*. Recuperado el 02 de Octubre de 2018, de [http://www.orin.com/access/sip\\_puff/](http://www.orin.com/access/sip_puff/)
- Ortega, C. (08 de Julio de 2018). *Tecnología asistiva para personas con discapacidad motora*. Recuperado el 05 de Octubre de 2018, de <http://www.youngmarketing.co/tecnologia-asistiva-para-personas-con-discapacidad-motora/>
- Project ALS. (Enero de 2018). *Sobre la enfermedad de ALS*. Obtenido de <https://www.projectals.org/about-als/>
- Tobii. (14 de Octubre de 2015). *Grupo Tobii*. Recuperado el 02 de Octubre de 2018, de <https://www.tobii.com/group/about/>
- Tobii Dynavox. (08 de Febrero de 2016). *Support for I-Series Classic*. Recuperado el 05 de Octubre de 2018, de <http://www2.tobiidynavox.co.uk/support/i-series-classic/>
- Torres Santos, E. R., & Bucheli Naranjo, J. L. (21 de julio de 2017). Construcción y modelación de un módulo didáctico de muro Trombe mediante CFD para fomentar los estudios de la calefacción solar pasiva. Quito, Pichincha, Ecuador: Escuela Politécnica Nacional. Obtenido de <http://bibdigital.epn.edu.ec/handle/15000/17509>
- UCSF Health. (2018). *Treatment ALS*. Recuperado el 05 de Octubre de 2018, de <https://www.ucsfhealth.org/conditions/als/treatment.html>
- Univisión-Noticias. (01 de Diciembre de 2014). *Educación Especial*. Recuperado el 05 de Octubre de 2018, de [¿Qué es la Tecnología de Asistencia?: https://www.univision.com/noticias/educacion-especial/que-es-la-tecnologia-de-asistencia](https://www.univision.com/noticias/educacion-especial/que-es-la-tecnologia-de-asistencia)

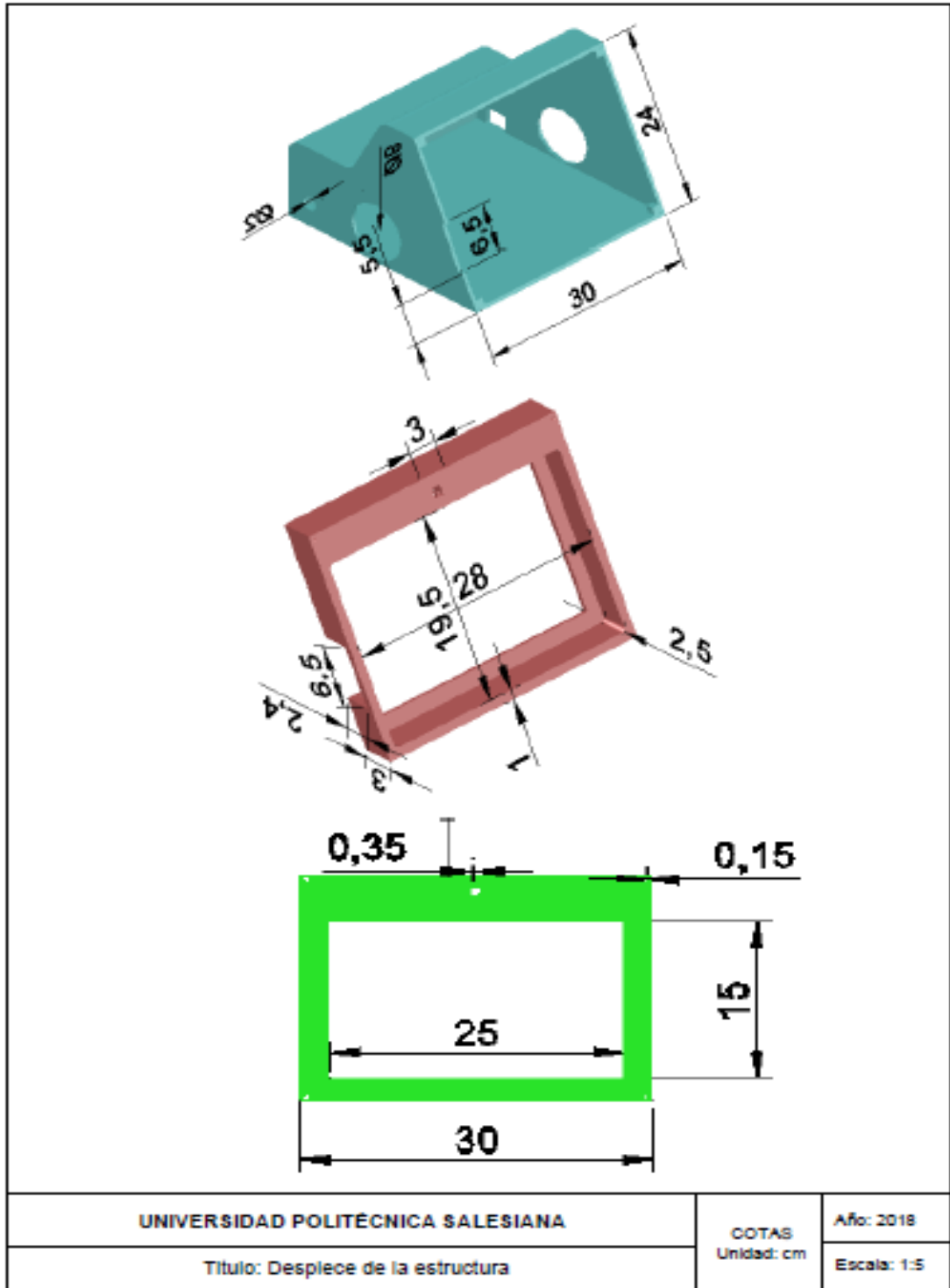


Wang, Y.-L., W.Y.Su, A., Han, T.-Y., Lin, C.-L., & Hsu, L.-C. (2015). EMG BASED REHABILITATION SYSTEMS – APPROACHES FOR ALS PATIENTS IN. *IEEE Explore Digital Library*, 1-6. doi:10.1109/ICME.2015.7177398

## **ANEXOS**

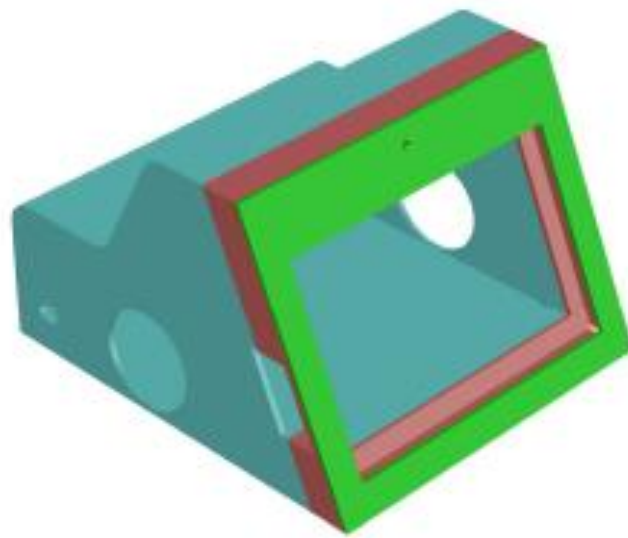
## Anexo A: Diseño de la estructura del prototipo en software AutoCAD

Estructura separada en tres partes con sus respectivas dimensiones como se puede apreciar.



Elaborado por: Luis Collaguazo

Estructura completa para la implementacion de los dispositivos electrónicos



<b>UNIVERSIDAD POLITÉCNICA SALESIANA</b>	COTAS	Año: 2018
Titulo: Estructura final	Unidad: cm	Escala: 1:5

Elaborado por: Luis Collaguazo

## Anexo B: Software de programación Python - Tkinter

**Código de programación del diccionario de palabras:** El diccionario contiene las 14 palabras por cada letra del abecedario, son las más empleadas en el lenguaje español y ayudan a una escritura rápida del paciente.

```
def dicPalabras():  
  
    dic = {  
        "a": ["acompañar", "ayuda", "aquel", "amigo", "apagar", "abajo", "aquí",  
              "adelante", "alimento", "ahora", "antes", "abierto", "además", "afectado"],  
        "b": ["baño", "basura", "buscar", "bonita", "bienestar", "beneficio", "bajar",  
              "bien", "beber", "bueno", "buenos", "base", "beso", "bello"],  
        "c": ["casa", "cómodo", "cama", "cabeza", "como", "casi", "comunicar",  
              "cosas", "cual", "con", "cualquier", "cambio", "cuerpo", "cuarto"],  
        "d": ["del", "desde", "donde", "después", "duro", "de", "durante",  
              "deseo", "difícil", "desarrollo", "días", "decir", "depresión", "desesperar"],  
        "e": ["enfermo", "enfermedad", "estas", "espero", "esos", "esclerosis", "ella",  
              "estoy", "es", "ela", "esfuerzo", "existe", "efecto", "educación"],  
        "f": ["fuerza", "facilidad", "futuro", "fecha", "fiebre", "forma", "final",  
              "familia", "furia", "frustración", "falso", "favorito", "famoso", "fantástico"],  
        "g": ["ganar", "género", "generación", "gracias", "grande", "gustar", "general",  
              "gente", "grandioso", "guiar", "girar", "gusto", "generoso", "guerra"],  
        "h": ["hay", "hola", "humildad", "hogar", "hombre", "hecho", "habitación",  
              "hoy", "herir", "hasta", "hace", "hambre", "hora", "historia"],  
        "i": ["identidad", "idea", "incapacidad", "interesante", "incendio", "igualdad", "importante",  
              "imposible", "inicio", "insistir", "informar", "intelectual", "ilusión", "ignorar"],  
        "j": ["jamás", "junto", "jabón", "Jesús", "jarabe", "joder", "jueves",  
              "joven", "juzgar", "justo", "jurar", "justicia", "joya", "jugar"],  
        "k": ["kilo", "kilómetro", "kiko", "kiosco", "kiwi", "karate", "karaoke",  
              "kilopondio", "kilovatio", "kilovoltio", "keniano", "kungfu", "kojo", "king_kong"],  
        "l": ["luchar", "lejos", "lectura", "los", "logro", "lugar", "luego",  
              "lógico", "leyenda", "ley", "la", "lámpara", "luz", "lastimar"],  
        "m": ["madre", "mayor", "miedo", "mismo", "movimiento", "muchos", "mañana",  
              "mundo", "mientras", "muy", "más", "música", "muerte", "momento"],  
        "n": ["nacimiento", "nacionalidad", "necesario", "ninguno", "noche", "necesidad", "nuestro",  
              "nada", "nunca", "número", "nombre", "nuevo", "novedoso", "norma"],  
        "ñ": ["ñaño", "ñaña", "ñoño", "ñoña", "ñado", "nata", ""],  
        "o": ["observar", "obscuridad", "ocultar", "olvidar", "orgullosa", "ordenar", "organizar", "opinar",  
              "otros", "obras", "obligación", "oportunidad", "operación", "órgano", "ofrecer"],  
        "p": ["para", "paciente", "paciencia", "pedir", "preguntar", "poner", "puedo",  
              "por", "pero", "pasado", "pues", "personas", "padre", "problemas"],  
        "q": ["quitar", "quien", "querer", "que", "quiero", "quienes", "quedate",  
              "quieto", "Química", "quemar", "quejar", "quehacer", "quisiera", "Quito"],  
        "r": ["radical", "reacción", "realidad", "riesgo", "realizar", "rapidamente", "rápido",  
              "relación", "respeto", "responsable", "ropa", "razón", "respirar", "regresar"],
```

```

"s": ["saber ", "sostener ", "sorprender ", "serenidad ", "sentido ", "sillón ", "suave ",
      "situación ", "soy ", "superar ", "serie ", "sentir ", "salud ", "seguir "],
"t": ["tambien ", " tiempo ", "total ", "tranquilidad ", "trabajo ", "todavía ", "todos ",
      "tener ", "tanto ", "trato ", "tarde ", "tampoco ", "tomar ", "temor "],
"u": ["union ", "utilizar ", "unidades ", "una ", "urgente ", "universal ", "universidad ",
      "un ", "ubicación ", "uniforme ", "umbral ", "único ", "usar ", "útil "],
"v": ["valiente ", "vecina ", "vejez ", "vehiculo ", "ventilador ", "vitamina ", "voluntad ",
      "ver ", "veces ", "verdad ", "ventana ", "visto ", "valor ", "vivienda "],
"w": ["whisky ", "walter ", "washington ", "wikipedia ", "Wi-Fi ", "whatsapp ", "web ",
      "wincha ", "windows ", "Wendy ", "Willian ", "Wilfrido ", "watt ", "wafle "],
"x": ["xenofobia ", "xerografía ", "xilema ", "xilofón ", "xilofonista ", " ", " ",
      " ", " ", " ", " ", " ", " ", " "],
"y": ["yugular ", "yerno ", "yeso ", "yema ", "yadira ", "yate ", "yoga ",
      "yodo ", "yerba ", "yarda ", "yoyo ", "yambo ", "yacente ", "yunque "],
"z": ["zapato ", "zaragoza ", "zoológico ", "zanahoria ", "zumbido ", "zinc ", "zona ",
      "zamora ", "zona ", "zarpozo ", "zurdo ", "zapote ", "zambullir ", "zumbar "],

}
return dic

```

Elaborado por: Luis Collaguazo

- **Teclado virtual:** La simulación del teclado virtual está realizado en Python-Tkinter, haciendo uso de varias librerías y métodos como se puede apreciar en el código a continuación.

```

#!/usr/bin/python3 #Este comando se usa para hacer ejecutable el archivo
from tkinter import *
from diccionarioPalabras import *
import os

teclado = Tk() # metodo constructor de Tkinter
teclado.title("TECLADO VIRTUAL") # comando para dar nombre a la ventana
teclado ["bg"] = '#00FFFF' # Comado para dar color de la ventana
teclado.geometry("1280x700+0+0") # (ancho, alto, x, y) # comando para dimencionar la ventana y su ubicación

# declaramos variables globales a usar, cuando estas parten del inicio
sonido = StringVar()
dato = 0
auxCont = 0
filaAux = 2
columAux = 0
tecla = 0
contador = 0
auxContador = 0

```

```

def diccionario(letraRx): # Inicio y forma de crear un metodo

    global auxContador # Asignacion de variables globales
    ancho = 50
    alto = 85
    cont = 0 # Asignacion de variables locales
    letraRx = letraRx.lower() # El dato que llega del teclado convierte de mayúsculas a minúsculas
    auxContador += 1 # Contador para el bloque del diccionario de palabras cuando es mayor que 1
    if teclaPulsada in claveDic and auxContador == 1: # Si la tecla esta en clave entonces obtengo las palabras
        palabra = list(dic.get(teclaPulsada)) # Obtengo todas las palabras que pertenecen a cada una de las claves
        for ubicar in palabra:
            pulso = lambda dicTx = ubicar: metodo_escribirDic(dicTx)

            if ubicar != "@": # Ubico las palabras en los botones que se crea
                Button(teclado, text= ubicar,
                    width=10, height=1,
                    font = ('arial', 17, 'bold'),
                    bg="#33FF8A", fg="#000000",
                    activebackground="#6633ff",
                    activeforeground="#FFFFFF",
                    relief = "raised",
                    padx=4, pady=4, bd=4,
                    command = pulso).place(x=ancho, y=alto)

        # Ubicacion de los botones
        cont += 1
        if cont == 1:
            ancho = 220
        if cont == 2:
            ancho = 390
        if cont == 3:
            ancho = 560
        if cont == 4:
            ancho = 730
        if cont == 5:
            ancho = 900
        if cont == 6:
            ancho = 1070
        if cont == 7:
            ancho = 50
            alto = 150
        if cont == 8:
            ancho = 220
            alto = 150
        if cont == 9:
            ancho = 390
            alto = 150
        if cont == 10:
            ancho = 560
            alto = 150
        if cont == 11:
            ancho = 730
            alto = 150
        if cont == 12:
            ancho = 900
            alto = 150
        if cont == 13:
            ancho = 1070
            alto = 150

```

```

else:
    if teclaPulsada == "*": # Caracter para habilitar los botones de diccionarios de palabras
        auxContador = 0

    for i in range(0,14): # Genera un rango de 14 caracteres para deshabilitar los botones
        Button(teclado, text= " ", state="disable", # Deshabilita los buttons
               width=10, height=1,
               font = ('arial', 17),
               bg="#33FF8A", fg="#ffffff",
               activebackground="#ffffff",
               activeforeground="#000000",
               relief = "raised",
               padx=4, pady=4, bd=4).place(x=ancho, y=alto)
        # Inicia el contador hasta el rango 14

        cont += 1
        if cont == 1:
            ancho = 220
        if cont == 2:
            ancho = 390
        if cont == 3:
            ancho = 560
        if cont == 4:
            ancho = 730
        if cont == 5:
            ancho = 900
        if cont == 6:
            ancho = 1070
        if cont == 7:
            ancho = 50
            alto = 150
        if cont == 8:
            ancho = 220
            alto = 150
        if cont == 9:
            ancho = 390
            alto = 150
        if cont == 10:
            ancho = 560
            alto = 150
        if cont == 11:
            ancho = 730
            alto = 150
        if cont == 12:
            ancho = 900
            alto = 150
        if cont == 13:
            ancho = 1070
            alto = 150

def metodo_escribirDic(dicRx): # Método que recibe palabras del diccionario y escribe
    global contador, auxContador
    if contador > 0:
        y = entrada.index(END)
        z = y - contador
        entrada.delete(z,END)
        entrada.insert(END, dicRx)
        contador = 0
        auxContador = 0

    else:
        entrada.insert(END, dicRx)

def metodo_escribir(tecla_Rx): # Método que recibe la letra pulsada del teclado principal
    global contador

    if tecla_Rx == "l a\nMayus":
        mayuscula()

```



```
elif tecla_Rx == "vozM" or tecla_Rx == "VOZM":
    voz = sonido.get()
    os.system('espeak -ves "{0}" .format (voz)) # Para el idioma ingles cambio -ven por -ves
```

```
elif tecla_Rx == "Borrar" or tecla_Rx == "BORRAR":
    borrar = entrada.index("end") # Ubica el cursor al final del texto
    entrada.delete((borrar)-1) # Borra texto desde el final hacia atras
    contador = 0
    diccionario("✱")
    auxContador = 0
```

```
elif tecla_Rx == "Borrar\nTodo" or tecla_Rx == "BORRAR\nTODO":
    entrada.delete(0, END)
    contador=0
    diccionario("✱")
    auxContador = 0
```

```
elif tecla_Rx == "Tab" or tecla_Rx == "TAB":
    entrada.insert(END, "    ")
```

```
else:
    if tecla_Rx != "SPACE":
        contador += 1
        entrada.insert(END, tecla_Rx)
        diccionario(tecla_Rx)
    else:
        entrada.insert(END, ' ')
        contador = 0
        diccionario("✱")
```

```
def minuscula():
    var_Fila = 3
    var_Columna = 0
    letras = [
        'Q', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0', '=', '{', '}', 'Borrar',
        'Tab', 'q', 'w', 'e', 'r', 't', 'y', 'u', 'i', 'o', 'p', '+', '(', ')', 'Borrar\nTodo',
        'l r a\nMayus', 'a', 's', 'd', 'f', 'g', 'h', 'j', 'k', 'l', 'ñ', '*', '[', ']', 'vozM',
        'Atras', 'z', 'x', 'c', 'v', 'b', 'n', 'm', ',', '.', ':', ';', '-', '=', 'vozM',
        'SPACE'
    ]
```

```
for tecla in letras:
    command = lambda tecla_Tx=tecla: metodo_escribir(tecla_Tx) # Comando lambda toma el dato y envia a
    if tecla != "SPACE":
        Button(teclado, text = tecla,
                width=5, height=3,
                font = ('arial', 18, 'bold'),
                bg="#5dade2", fg="#000000",
                activebackground="#6633ff",
                activeforeground="#ffffff",
                relief = "raised",
                padx=4, pady=4, bd=4, # ancho de bordes
                command = command).grid(row=var Fila, column=var Columna)
```

```

if tecla == "Borrar\nTodo" or tecla == "lr a\nMayus" or tecla == "Borrar" or tecla == "Tab" or tecla == "vozM":
    Button(teclado, text= tecla,
           width=5, height=3,
           font = ('arial', 17, 'bold'),
           bg="#5dade2", fg="#00008b",
           activebackground="#6633ff",
           activeforeground="#ffffff",
           relief = "raised",
           padx=4, pady=4, bd=4,
           command = command).grid(row=var_Fila, column=var_Columna)

if tecla == "Atras":
    Button(teclado, text= tecla,
           width=5, height=3,
           font = ('arial', 17, 'bold'),
           bg="red", fg="white",
           activebackground="red",
           activeforeground="black",
           relief = "raised",
           padx=4, pady=4, bd=4,
           command = teclado.destroy).grid(row=var_Fila, column=var_Columna)

if tecla == "SPACE":
    Button(teclado, text= tecla,
           width=50, height=2,
           font = ('arial', 17, 'bold'),
           bg="#5dade2", fg="#000000", # bg=fondo, fg=letra
           activebackground="#6633ff",
           activeforeground="#ffffff",
           relief = "raised",
           padx=3, pady=3, bd=3,
           command = command).grid(row=15, columnspan=20)

if tecla == "vozM":
    Button(teclado, text= tecla,
           width=5, height=7,
           font = ('arial', 17, 'bold'),
           bg="blue", fg="#000000",
           activebackground="#6633ff",
           activeforeground="#ffffff",
           relief = "raised",
           padx=3, pady=3, bd=3,
           command = command).place(x=1192. v = 420)

#Ordenando los botones del teclado
var_Columna += 1

if var_Columna >14 and var_Fila ==3:
    var_Columna = 0
    var_Fila +=1

if var_Columna >14 and var_Fila ==4:
    var_Columna = 0
    var_Fila +=1

if var_Columna > 14 and var_Fila ==5:
    var_Columna =0
    var_Fila +=1

if var_Columna > 14 and var_Fila ==6:
    var_Columna =0
    var_Fila +=1

if var_Columna > 14 and var_Fila ==7:
    var_Columna =0
    var_Fila +=1

```

```

def mayuscula():
    var_FilaM = 3
    var_ColumnaM = 0

    letras = [
        '~', '|', '|', '@', '#', '$', '%', '^', '&', 'z', '?', '<', '>', '!', 'Borrar',
        'Tab', 'q', 'w', 'e', 'r', 't', 'y', 'u', 'i', 'o', 'p', '"', '"', 'Borrar\nTodo',
        'r a\nMinus', 'a', 's', 'd', 'f', 'g', 'h', 'j', 'k', 'l', 'ñ', 'ó', '≠', '/', 'vozM',
        'Atras', 'z', 'x', 'c', 'v', 'b', 'n', 'm', '←', '⌕', '£', '₹', '≤', '≥', 'vozM',
        'SPACE'
    ]

    for teclaM in letras:
        teclaM = teclaM.upper() # El método upper permite convertir de minúsculas a mayúsculas
        commandoM = lambda tecla_Tx=teclaM: metodo_escribir(tecla_Tx)

        if tecla1 != "SPACE": # and tecla1 != "BLOQ\nMAYUS" and tecla1 != "Borrar\nTODO" and tecla1 != "VOZF" and tecla1 != "VOZM":
            Button(teclado, text = tecla1,
                    width=5, height=3,
                    font = ('arial', 17, 'bold'),
                    bg="#5dade2", fg="#000000",
                    activebackground="#6633ff",
                    activeforeground="ffffff",
                    relief = "raised",
                    padx=4, pady=4, bd=4, # ancho de bordes
                    command = commandoM).grid(row=var_FilaM, column=var_ColumnaM)

        if tecla1 == "Borrar\nTODO" or tecla1 == "VOZM" or tecla1 == "Borrar" or tecla1 == "r A\nMINUS" or tecla1 == Button(teclado,
                    text= teclaM,
                    width=5, height=3,
                    font = ('arial', 17, 'bold'),
                    bg="#5dade2", fg="#00008b",
                    activebackground="#6633ff",
                    activeforeground="ffffff",
                    relief = "raised",
                    padx=4, pady=4, bd=4,
                    command = commandoM).grid(row=var_FilaM, column=var_ColumnaM)

        if tecla1 == "ATRAS":
            Button(teclado, text= teclaM,
                    width=5, height=3,
                    font = ('arial', 17, 'bold'),
                    bg="red", fg="white",
                    activebackground="red",
                    activeforeground="black",
                    relief = "raised",
                    padx=4, pady=4, bd=4,
                    command = teclado.destroy).grid(row=var_FilaM, column=var_ColumnaM)

        if tecla1 == "SPACE":
            Button(teclado, text= teclaM,
                    width=50, height=2,
                    font = ('arial', 17, 'bold'),
                    bg="#5dade2", fg="#000000", # bg=fondo, fg=letra
                    activebackground="#6633ff",
                    activeforeground="ffffff",
                    relief = "raised",
                    padx=3, pady=3, bd=3,
                    command = commandoM).grid(row=15, colspan=20)

```

```

if tecla1 == "VOZM":
    Button(teclado, text= teclaM,
           width=5, height=7,
           font = ('arial', 17, 'bold'),
           bg="blue", fg="#000000",
           activebackground="#6633ff",
           activeforeground="#ffffff",
           relief = "raised",
           padx=3, pady=3, bd=3,
           command = comandoM).place(x=1192, y = 420)

var_ColumnaM += 1
if var_ColumnaM >14 and var_FilaM ==3:
    var_ColumnaM = 0
    var_FilaM +=1

if var_ColumnaM >14 and var_FilaM ==4:
    var_ColumnaM = 0
    var_FilaM +=1

if var_ColumnaM > 14 and var_FilaM ==5:
    var_ColumnaM =0
    var_FilaM +=1

if var_ColumnaM > 14 and var_FilaM ==6:
    var_ColumnaM =0
    var_FilaM +=1

if var_ColumnaM > 14 and var_FilaM ==7:
    var_ColumnaM =0
    var_FilaM +=1

```

```

etiqueta1= Label(teclado, text='', height=1, font = ('arial', 15, 'bold'), bg = '#00FFFF').grid(row = 0, column=0) # Etiquetas para dar espacios
entrada = Entry(teclado, textvariable=sonido, width=83, font = ('arial', 20, 'bold'), fg = 'black' ) # Comando para la entrada de texto
entrada.focus_set() # Comando para habilitar el cursor caso contrario no es visible el cursor
entrada.grid(row=1, columnspan=20) #.grid permite ubicar y cargar el texto entry  bg="#00ff0", fg="#ffffff",

```

```

etiqueta2 = Label(teclado, text="", height = 7, font = ('arial', 15, 'bold'), bg = '#00FFFF').grid(row =2 , column=0)

```

```

minuscula() # Inicializo el teclado con minusculas

```

```

teclado.mainloop() # Cierro el constructor de la ventana

```

Elaborado por: Luis Collaguazo

- **Mensajes predefinidos:** Estos mensajes permiten agilizar la comunicación y pueden ser cambiados de acuerdo a las necesidades del paciente.

```

#!/usr/bin/python3
# -*- coding: utf-8 -*-
from tkinter import *
import os
#configuramos la ventana
ventanaMensajes = Tk()
ventanaMensajes.title("MENSAJES PREDEFINIDOS")
ventanaMensajes.geometry("1280x700+0+0")
ventanaMensajes.config(bg = "#6699ff")

```

```

def metodo_reproducir(letraRx):
    if letraRx != boton21:
        os.system('espeak -ves "{0}"'.format (letraRx))
boton1 = "Hola amigo, muchas gracias por visitarme"
boton2 = "Mi nombre es Luis, soy un enfermo con ela"
boton3 = "Padecer la enfermedad de ela es muy difícil"
boton4 = "Sufro de esta enfermedad desde los 40 años"
boton5 = "Antes de padecer esta enfermedad era muy feliz"
boton6 = "Me gustaba salir con mi familia a divertirme"
boton7 = "Esta enfermedad cambió mi vida totalmente"
boton8 = "Gracias al apoyo de mi familia sigo adelante"
boton9 = "Gracias a este dispositivo puedo comunicarme"
boton10 = "Gracias por visitarme, espero verte pronto"
boton11 = "Necesito mi medicina, ayuda por favor"
boton12 = "Deseo salir ayudenme por fovor"
boton13 = "Necesito hacerme el aseo por favor ayuda "
boton14 = "Deseo ver la tele, encienda por favor"
boton15 = "Quiero alimentarme, ayuda por favor"
boton16 = "Quiero acostarme ayudenme por favor"
boton17 = "Necesito que me revise el médico, me siento mal "
boton18 = "Siento frío, encienda el calefactor por favor"
boton19 = "Siento calor, encienda el ventilador por favor"
boton20 = "Quiero descansar dejenme solo por favor"
boton21 = "ATRAS"

lista = [ boton1, boton2, boton3, boton4, boton5, boton6, boton7, boton8, boton9, boton10, boton11,
          boton12, boton13, boton14, boton15, boton16, boton17, boton18, boton19, boton20, boton21
          ]

ancho = 5
ancho2=715
alto = 10
contador = 0

for letra in lista:
    pulso = lambda letraTx = letra: metodo_reproducir(letraTx)

    if letra != boton21:
        Button(ventanaMensajes, text = letra,
               font = ('arial', 17, 'bold'), bd = 6, command = pulso,
               width= 40, height=2, bg = "#00ffff", fg = "black",
               activebackground="#ff6633",
               activeforeground="white").place(x=ancho, y = alto)

    if letra == boton21:
        Button(ventanaMensajes, text = letra,
               font = ('arial', 15, 'bold'), bd = 5, command = ventanaMensajes.destroy,
               bg = "#99ff33", fg = "black", width= 15, height =2,
               activebackground="#ff6633",
               activeforeground="white").place(x=540, y=628)

    contador += 1

    if contador == 1:
        alto = 70
    if contador == 2:
        alto = 130
    if contador == 3:
        alto = 190
    if contador == 4:
        alto = 250
    if contador == 5:
        alto = 310
    if contador == 6:
        alto = 370
    if contador == 7:
        alto = 430
    if contador == 8:
        alto = 490
    if contador == 9:
        alto = 550
    if contador == 10:
        alto = 10
        ancho = ancho2

```

```

if contador == 11:
    alto = 70
    ancho = ancho2
if contador == 12:
    alto = 130
    ancho = ancho2
if contador == 13:
    alto = 190
    ancho = ancho2
if contador == 14:
    alto = 250
    ancho = ancho2
if contador == 15:
    alto = 310
    ancho = ancho2
if contador == 16:
    alto = 370
    ancho = ancho2
if contador == 17:
    alto = 430
    ancho = ancho2
if contador == 18:
    alto = 490
    ancho = ancho2
if contador == 19:
    alto = 550
    ancho = ancho2
if contador == 20:
    alto = 610
    ancho = ancho2

```

```
ventanaMensajes.mainloop()
```

Elaborado por: Luis Collaguazo

- **Menú principal:** Este menú contiene todas la funciones de comunicación como son: teclado virtual, mensajes predefinidos, envío de notificaciones y reproductor multimedia.

```

#!/usr/bin/python2
#encoding: utf-8 # comando para tildes y otras funciones
from Tkinter import *
import urllib2 # Libreria para enviar notificación
import json #(sincronización con el servidor de la nube) para la notificación
import os # Módulo para trabajar con varios métodos de Python

#configuramos la ventana
ventanaMenu = Tk()
ventanaMenu.title("MENU")
ventanaMenu.geometry("1280x700+0+0")
ventanaMenu.config(bg = "#00FFFF")

x = "5cee70f3b3af8e4977a4f0413d38d4494f7ef55b" # Token o clave que me dan al momento de crear la cuenta
y = "CanalLuis" # Canal de Pushetta
z = "¡¡¡ MENSAJE DE EMERGENCIA !!! (PACIENTE - ELA)" #Mensaje a enviar

```

```

def envioN(token, channel, message): #Metodo que recibe los parametros de x, y i z
    dato = {"body": message, "message_type": "text/plain"}
    req = urllib2.Request('http://api.pushetta.com/api/pushes/{0}'.format(channel))
    req.add_header('Content-Type', 'application/json')
    req.add_header('Authorization', 'Token {0}'.format(token))
    respuesta= urllib2.urlopen(req, json.dumps(dato)) # Hasta aquí el envío de notificación

def Metodo_teclado(): # Método para el llamado del teclado
    os.system("/usr/bin/python3 tecladofinal.py") # Comando para ejecutar archivos con Python

def Metodo_notificacion(): # Método para el llamado de notificación
    envioN(x,y,z) # Envío de parametros de la notificacion
    os.system('espeak -ves "Mensaje Enviado") # reproducción del mensaje enviado

def Metodo_mensaje(): # Método de mensajes predefinidos
    os.system("/usr/bin/python3 mensajesP.py")

def Metodo_musica(): # Método para la reproductor multimedia
    os.popen("smplayer") #Comando para abrir la aplicaion

# cargamos la imagen
#imagen1 = PhotoImage(file = "E:/Ejemplos_Python/UPS.jpg")
imagenUPS = PhotoImage(file = "/home/luis/Escritorio/TesisFinal/UPS1.png") # Comando para cargar la imagen a una variable
imagenELA = PhotoImage(file = "/home/luis/Escritorio/TesisFinal/ELA.png")
imagenTeclado1 = PhotoImage(file = "/home/luis/Escritorio/TesisFinal/teclado1.png")
imagenTeclado2 = PhotoImage(file = "/home/luis/Escritorio/TesisFinal/teclado2.png")
imagenTeclado3 = PhotoImage(file = "/home/luis/Escritorio/TesisFinal/teclado3.png")
imagenTeclado4 = PhotoImage(file = "/home/luis/Escritorio/TesisFinal/teclado4.png")

botonUPS = Button(ventanaMenu, image = imagenUPS, padx=10, pady=10, bd=0),place(x=5, y=5) # código para cargar la imagen en el button
botonELA = Button(ventanaMenu, image = imagenELA, padx=10, pady=10, bd=6, bg = 'black', activebackground='black').place(x=350, y=15)
etiquetaTitulo = Label(ventanaMenu, text='PROYECTO DE INVESTIGACIÓN: PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA',
    font = ('arial', 17, 'bold'), bg = '#00FFFF', fg = 'black').place (x =45 , y =365)
etiquetaAutor = Label(ventanaMenu, text='Autor: Luis Collaguazo', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'black').place (x =505 , y =420)

etiquetaTeclado = Label(ventanaMenu, text='Teclado Virtual', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =80 , y =480)

etiquetaMensajes = Label(ventanaMenu, text='Mensajes Predefenidos', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =345 , y =480)

etiquetaNotificacion = Label(ventanaMenu, text='Notificación', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =720 , y =480)

etiquetaMusica = Label(ventanaMenu, text='Reproductor Multimedia', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =970 , y =480)

etiquetaMensajes = Label(ventanaMenu, text='Mensajes Predefenidos', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =345 , y =480)

etiquetaNotificacion = Label(ventanaMenu, text='Notificación', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =720 , y =480)

etiquetaMusica = Label(ventanaMenu, text='Reproductor Multimedia', font = ('arial', 17, 'bold'),
    bg = '#00FFFF', fg = 'blue').place (x =970 , y =480)

```

```
botonTeclado = Button(ventanaMenu, image = imagenTeclado1, command = Metodo_teclado ,
                      font = ('arial', 25, 'bold'), bg = 'gray', activebackground="blue",
                      padx=10, pady=10, bd=10).place(x =50 , y =520)

botonMensajes = Button(ventanaMenu, image = imagenTeclado3, command = Metodo_mensaje,
                      font = ('arial', 25, 'bold'), bg = 'gray', activebackground="blue",
                      padx=10, pady=10, bd=10).place(x =370 , y =520)

botonNotificacion = Button(ventanaMenu, image = imagenTeclado2, command = Metodo_notificacion,
                           font = ('arial', 25, 'bold'), bg = 'gray', activebackground="blue",
                           padx=10, pady=10, bd=10).place(x =680 , y =520)
botonMusic = Button(ventanaMenu, image = imagenTeclado4, command = Metodo_musica,
                   font = ('arial', 25, 'bold'), bg = 'gray', activebackground="blue",
                   padx=10, pady=10, bd=10).place(x =1000 , y =520)

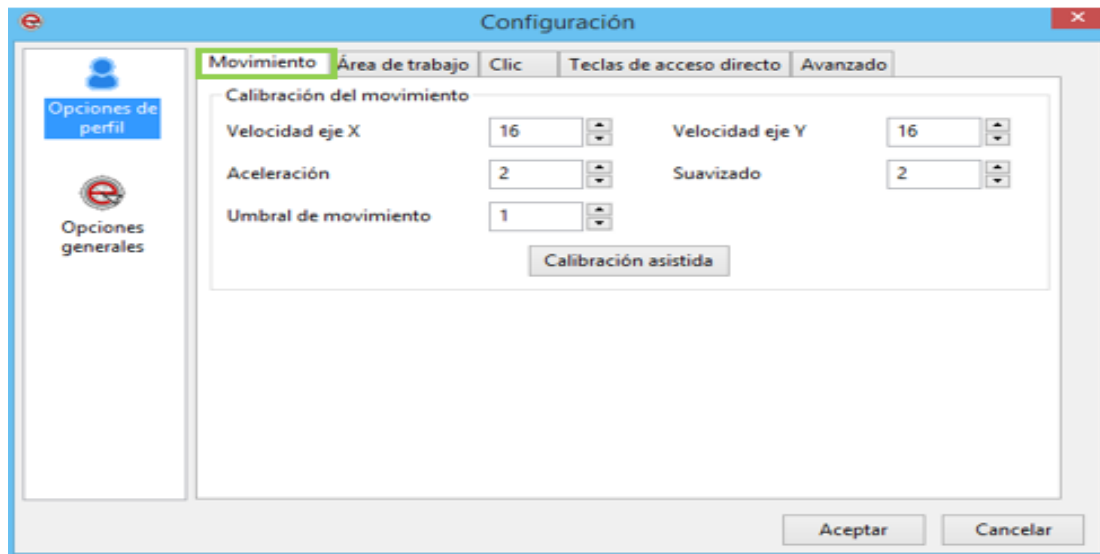
botonSalir = Button(ventanaMenu, text = "SALIR", command = ventanaMenu.destroy, height=3, width=10,
                   font = ('arial', 25, 'bold'), bg = 'red', fg = 'white',
                   activebackground="red", activeforeground="black",
                   padx=10, pady=10, bd=5).place(x =1060 , y = 10)

ventanaMenu.mainloop() # Cierro la creacion de la ventana principal
```



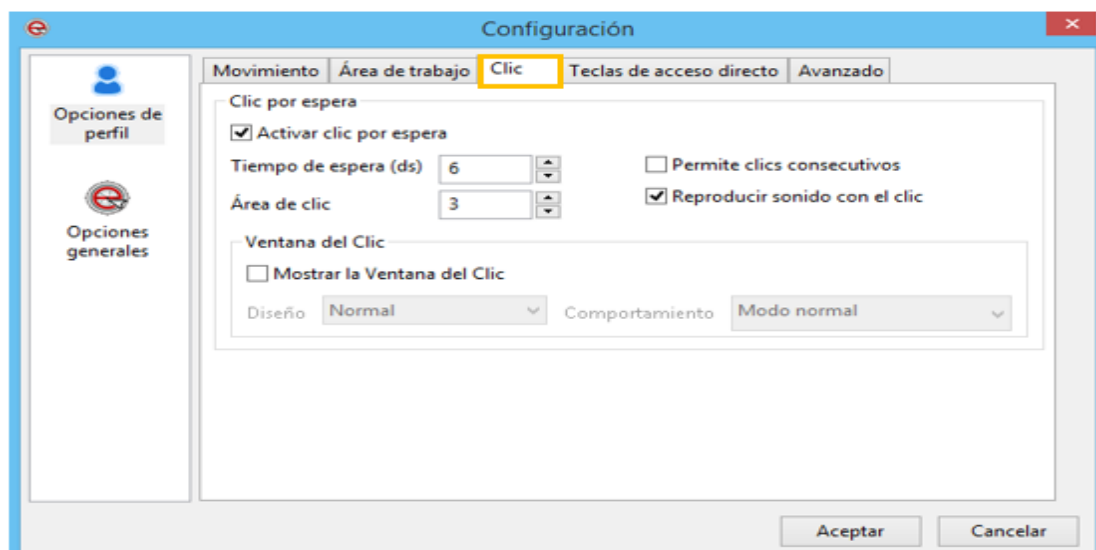
## Anexo C: Configuración de la aplicación de Eviacam para el seguimiento facial

Al ejecutar la aplicación de Eviacam e ir a la opción configuraciones, se calibra la velocidad y el movimiento del cursor en la pantalla, esta calibración se hace de acuerdo a la capacidad de movimiento de la cabeza de cada usuario, en la imagen a continuación se muestra los valores de dicha calibración.



Elaborado por: Luis Collaguazo


Seguidamente se calibra el tiempo de espera del clic, el clic hace cuando el cursor este fijo sobre un botón, en la imagen a continuación se muestra los valores de calibración.



Elaborado por: Luis Collaguazo





**Anexo D: Cotización de los dispositivos comerciales para la comunicación de los pacientes con la enfermedad de ELA.**

Cotización del dispositivo EyeMobile Mini de la empresa Tobii Dynavox

 <p>ASISTRONIC Tecnologías de Asistencia www.asistronic.com</p>	<p>Asistronic S.A.S Nit. 901.144.170-0 Cra.10 No.97A-13 Ofic. 202 Torre B Bogotá-Colombia PBX: (571) 7422737-7592504 Cel: (57) 3103627093</p>


Estimado cliente  
Luis Collaguazo  
Pillaro, Provincia del Tungurahua, Ecuador

Fecha: 11/10/2018  
Vendedor: Diego Gil  
Cotización ES-111018-01  
Moneda: Dolar Americano

Código	Imagen	Descripción	Marca	Patología	Q	Valor unitario	Valor total
111081		Sistema Tobii Dynavox EyeMobile Mini de comunicación y control por mirada. Incluye pantalla de 12", Sensor Tobii Dynavox Pceye Mini, Bracket EyeMobile, Software Tobii Communicator 5	Tobii	Varias	1	\$3.914	\$3.914
12005079		Montura soporte abrazadera RehADAPT de múltiples posiciones para sillas de ruedas y camas	Connect-It	Varias	1	\$465	\$465
520109		Maletín viajero protector original de Tobii Dynavox	Tobii	Varias	1	\$116	\$116
		Envío a Ecuador. Incluye soporte técnico por un año	Asistronic	Varias	1	\$264	\$264
						<b>Subtotal</b>	<b>\$4.759</b>



Elaborado por: Luis Collaguazo

## Cotización del dispositivo Tobii I15+ Plus de la empresa Tobii Dynavox

 ASISTRONIC Tecnologías de Asistencia www.asistronic.com	Asistronic S.A.S Nit. 901.144.170-0 Cra.10 No.97A-13 Ofic. 202 Torre B Bogotá-Colombia PBX: (571) 7422737-7592504 Cel: (57) 3103627093
--	--

Estimado cliente  
 Luis Collaguazo  
 Pillaro, Provincia del Tungurahua, Ecuador

Fecha: 07/10/2018  
 Vendedor: Diego Gil  
 Cotización ES-071018-01  
 Moneda: Dólar Americano

Código	Imagen	Descripción	Marca	Patología	Q	Valor unitario	Valor total
113163		Comunicador Tobii I-15+ Gaze Interaction con Software Tobii Communicator 5 y Tobii Sono Suite. Incluye monitor integrado antigolpes, cero ruido, posibilidad de esterilizarse	Tobii	Varias	1	\$14.150	\$14.150
12005079		Montura soporte abrazadera RehADAPT de múltiples posiciones para sillas de ruedas y camas	Connect-It	Varias	1	\$465	\$465
520109		Maletín viajero protector original de Tobii Dynavox	Tobii	Varias	1	\$116	\$116
		Entrega y capacitación en Ecuador, cuatro horas con un Ingeniero de la compañía. Incluye soporte técnico por un año	Asistronic	Varias	1	\$264	\$264
						<b>Subtotal</b>	<b>\$14.996</b>
						<b>Iva del 19%</b>	<b>\$0</b>
						<b>Total Compra</b>	<b>\$14.996</b>

Para entrega y capacitación en Ecuador con un Ingeniero de la compañía por cuatro horas sumar 1,050 USD

Elaborado por: Luis Collaguazo