

DESARROLLO DE UNA INTERFAZ BCI PARA LA COMUNICACIÓN ASÍNCRONA  
EN DISPOSITIVOS MÓVILES POR MEDIO DEL PARADIGMA P300

JEFFERSON STIVEN CORONADO PINZÓN  
JONNATHAN NICOLÁS SUÁREZ TORRES

REALIZADO CON LA ASESORÍA DE:

ÓSCAR MAURICIO ARIAS BALLÉN

FIDEL MAURICIO BONILLA

UNIVERSIDAD EL BOSQUE  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
DICIEMBRE, 2023

UNIVERSIDAD EL BOSQUE  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA

ÁREA DE ÉNFASIS: EQUIPOS MEDICOS

DESARROLLO DE UNA INTERFAZ BCI PARA LA COMUNICACIÓN ASÍNCRONA  
EN DISPOSITIVOS MÓVILES POR MEDIO DEL PARADIGMA P300

JEFFERSON STIVEN CORONADO PINZÓN  
JONNATHAN NICOLÁS SUÁREZ TORRES

REALIZADO CON LA ASESORÍA DE:

ÓSCAR MAURICIO ARIAS BALLÉN

FIDEL MAURICIO BONILLA

Página de Aprobación. Inclusión de Acta de grado.

## **NOTA DE SALVEDAD**

Según el artículo 37 del 14 de diciembre de 1989 del acuerdo 017, “La Universidad El Bosque, no se hace responsable de los conceptos emitidos por los investigadores en su trabajo, solo velará por el rigor científico, metodológico y ético del mismo en aras de la búsqueda de la verdad y la justicia”.

## **DEDICATORIA**

A mi madre Gladys la cual me apoyo durante cada parte de mi carrera en cada palabra de aliento e impulso en los momentos difíciles.

A mi madre Claudia, un pilar inquebrantable a lo largo de mi vida y trayectoria. Su constante presencia y apoyo incansable han iluminado cada paso que he dado.

A mi hermano Howarth quien siempre estuvo a mi lado durante las largas investigaciones.

A nuestras parejas Carlos y Natalia las cuales, gracias a sus mensajes de motivación, su apoyo incondicional y su gran entusiasmo nos impulsaron a llegar hasta acá.

## **AGRADECIMIENTOS**

Agradecerle a nuestro director de proyecto Óscar Arias por siempre apoyarnos en cada tarea, actividad y proceso para culminar este proyecto. A cada uno de los docentes de la facultad que nos enseñaron todos los conocimientos y habilidades para superar cada reto que sobrellevemos.

## RESUMEN

El presente proyecto se centró en el diseño e implementación de una interfaz cerebro-ordenador (BCI) basada en el paradigma P300, con el propósito de abordar diversas problemáticas identificadas en las interfaces BCI existentes. Estas problemáticas incluyen la eficiencia, precisión, adaptabilidad y operabilidad.

La interfaz como plataforma web permitió a pacientes comunicarse de forma asincrónica a través de dispositivos móviles, que además utilizó señales electroencefalográficas no invasivas con componentes de ondas P300 en respuesta a estímulos visuales presentados en un ordenador. La interfaz creada facilitó la visualización y procesamiento de las señales EEG, la configuración personalizada para cada usuario y el envío de mensajes compuestos por caracteres hacia un bot de la aplicación móvil Telegram.

Además, se utilizó el lenguaje de programación Python para el desarrollo y construcción de todos los algoritmos de análisis, filtrado, procesamiento, clasificación y entrenamiento de datos. A través de la identificación de eventos relacionados con potenciales P300 que representaran los caracteres individuales seleccionados por la persona a través de una matriz de estímulos visuales. Los resultados indicaron una precisión de más del 60% en la clasificación de señales EEG utilizando un algoritmo de entrenamiento incluido en el clasificador *OneRule* como elemento de *machine learning* integrado con procesamiento de señales por Transformada de Wavelet.

**Palabras Clave:** Comunicación móvil, Interfaz, Procesamiento de la información, Visualización

## **ABSTRACT**

This project focused on the design and implementation of a brain-computer interface (BCI) based on the P300 paradigm, with the purpose of addressing several issues identified in existing BCI interfaces. These issues include efficiency, accuracy, adaptability, and operability.

The interface as a web-based platform allowed patients to communicate asynchronously via mobile devices, which also used non-invasive electroencephalographic signals with P300 wave components in response to visual stimuli presented on a computer. The interface created facilitated the visualization and processing of EEG signals, the personalized configuration for each user and the sending of messages composed of characters to a bot of the Telegram mobile application.

In addition, the Python programming language was used for the development and construction of all data analysis, filtering, processing, classification and training algorithms. Through the identification of events related to P300 potentials that represented individual characters selected by the person through a matrix of visual stimuli. The results indicated an accuracy of more than 60% in the classification of EEG signals using a training algorithm included in the OneRule classifier as a machine learning element integrated with Wavelet Transform signal processing.

**Keywords:** Information processing, Interface, Mobile communication, Visualization.



## GLOSARIO DE TÉRMINOS

**Array:** Tipo de dato estructurado que permite almacenar datos del mismo tipo de forma secuencial, utilizado en los lenguajes de programación.

**BackEnd:** Servicios que maneja desde el lado del servidor, la lógica respecto a un sistema.

**Bot:** Aplicación que se ejecuta dentro de una aplicación de mensajería para interactuar como si fuera una persona real

**Endpoint:** Dirección de una API que genera una respuesta respecto a una solicitud.

**Framework:** Conjunto estructuras, librerías y estándares para la construcción de un sistema en *frontend* o en *backend*.

**FrontEnd:** Parte visual o informativa de sitio, plataforma o aplicación web.

**Lenguaje de programación:** Programación Informática (definición tomada de Tesauro de UNESCO)

**Puerto COM:** Interfaz de entrada o salida que permite conectar un dispositivo al computador

**Teorema de Nyquist:** Expresa la relación entre la frecuencia de muestreo y frecuencia de una señal medida

**Telegram:** Aplicación móvil y plataforma web de mensajería instantánea.

**URL:** Dirección de un recurso único en la web.

## LISTA DE SIMBOLOS

$T_1$  Periodo de muestreo a frecuencia de muestreo de 250 Hz

$T_2$  Periodo de muestreo a frecuencia de muestreo de 125 Hz

$N_V$  Número de muestras para tamaño de ventana

$N_d$  Número de muestras de desplazamiento para ventanas

$f_{cm}$  Frecuencia máxima de corte para filtro pasa banda

$H_{fn}$  Ecuación en forma canónica de filtro notch

$H_{fb}$  Ecuación en forma canónica de filtro pasa banda

## **LISTA DE ABREVIATURAS**

ACC: Análisis por correlación cruzada  
ADF: Análisis en el dominio de la frecuencia  
API: Interfaz de programación de aplicaciones  
BCI: Interfaz cerebro-ordenador  
EEG: Electroencefalográficas  
ERP: Potencial de eventos relacionados  
FFT: Transformada Rápida de Fourier  
Hz: Hertz (Unidad de Medida)  
ID: Identificación  
ms: Milisegundos  
PCA: Análisis de componentes principales  
URL: Localizador de recursos uniforme  
WT: Transformada de Wavelet

## TABLA DE CONTENIDOS

	Pág.
1. INTRODUCCIÓN .....	19
2. PLANTEAMIENTO DEL PROBLEMA .....	20
2.1 Antecedentes y Estado del Arte .....	20
Nivel Internacional .....	20
Nivel Nacional .....	21
2.2 Descripción y Formulación del Problema .....	23
2.3 Justificación .....	24
2.4 Objetivos .....	25
Objetivo General .....	25
Objetivos Específicos .....	26
2.5 Alcance y Limitaciones del Proyecto .....	26
3. MARCO DE REFERENCIA .....	27
3.1 Marco Teórico o Conceptual .....	27
Transformada de wavelet .....	27
Transformada de wavelet discreta .....	27
Familias de señales de wavelet .....	28
Ondas EEG y bandas asociadas .....	29
Paradigma P300 .....	30
Interfaz BCI .....	31
Algoritmo clasificador OneRule .....	32
3.2 Marco Legal o Normativo .....	33
4. DESARROLLO DEL PROYECTO DE GRADO .....	34
4.1 Requerimientos .....	34
4.1.1 Requerimiento General .....	34
4.1.2 Requerimientos funcionales .....	34
4.1.3 Requerimientos no funcionales .....	35
4.1.4 Requerimientos de calidad .....	35

4.1.5	Requerimientos de restricción .....	35
4.2	Metodología del Diseño .....	36
4.2.1	Diseño .....	36
4.2.2	Implementación .....	51
4.2.3	Validación .....	73
4.3	Descripción Técnica del Producto .....	73
	Modelo UML .....	73
	Diagrama de flujo de funcionamiento del sistema.....	76
	Modelo Entidad-Relación .....	77
5.	RESULTADOS Y ANÁLISIS DE RESULTADOS .....	79
5.1	Protocolo de ejecución de procedimientos .....	79
5.2	Características de los voluntarios .....	79
5.3	Características de las pruebas .....	80
5.4	Preparación y conexión de elementos a voluntarios .....	80
5.5	Visualización de señales .....	81
5.1	Relación de datos obtenidos .....	82
6.	CONCLUSIONES .....	90
7.	RECOMENDACIONES .....	91
8.	REFERENCIAS BIBLIOGRÁFICAS .....	92
	ANEXO A. Modelo de Arquitectura del sistema .....	95
	ANEXO B. Modelo Entidad – Relación de la base de datos.....	96
	ANEXO C. Manual de Usuario Tarjeta por Integración <i>Cyton</i> de OpenBCI .....	96
	ANEXO D. Manual de Usuario para plataforma web y administración de usuarios .....	100
	ANEXO E. Manual de Usuario interfaz de adquisición de señales.....	123
	ANEXO F. Repositorio de almacenamiento de código .....	125
	ANEXO G. Documentación para conexión y uso API Telegram .....	125
	ANEXO H. Plan de pruebas del sistema. ....	126

## LISTA DE TABLAS

Tabla	Pág.
Tabla 1. Características por formato de datos para selección para selección de técnica. ..	45
Tabla 2. Características de posición de electrodos para selección de modelo.....	45
Tabla 3. Características por librería base de uso para selección de procedimiento .....	46
Tabla 4. Características por técnica para detección de la onda .....	47
Tabla 5. Características por tipo de receptor para selección de modelo.....	47
Tabla 6. Características por técnica de clasificación para selección .....	48
Tabla 7. Características por tipo de filtro digital .....	48
Tabla 8. Características por framework basado en JavaScript para frontend .....	49
Tabla 9. Características por framework basado en Python para backend .....	49
Tabla 10. Características por voluntario .....	79
Tabla 11. características de las pruebas ejecutadas .....	80
Tabla 12. Resultados de experimentos para el voluntario A .....	83
Tabla 13. Resultados de experimentos para el voluntario B.....	84
Tabla 14. Resultados de experimentos para el voluntario C.....	85
Tabla 15. Relación de precisión del sistema para cada voluntario .....	87
Tabla 16. Plan de pruebas .....	126

## LISTA DE FIGURAS

Figura	Pág.
Figura 1. Diagrama de caja negra .....	37
Figura 2. Diagrama funcional basado en el diagrama de caja negra .....	37
Figura 3. Diagrama interno para módulo de procesamiento.....	39
Figura 4. Diagrama interno para módulo de filtro pasa banda y notch .....	39
Figura 5. Diagrama interno para módulo de transformada de wavelet.....	40
Figura 6. Diagrama interno para módulo de clasificación.....	41
Figura 7. Diagrama interno para módulo de identificación .....	41
Figura 8. Diagrama interno para módulo de visualización .....	42
Figura 9. Diagrama interno para módulo de comunicación.....	42
Figura 10. Diagrama interno para módulo tratamiento de configuraciones .....	43
Figura 11. Diagrama interno para módulo generador de estímulos.....	43
Figura 12. Formato Binario para tarjeta Cyton de OpenBCI obtenida de [16] .....	44
Figura 13. Mockups para la interfaz (a) Dashboard (b) Experiments (c) Settings (d) Communication.....	50
Figura 14. Vista principal Login de plataforma web .....	51
Figura 15. Diagrama de flujo de la validación de credenciales y autenticación de usuarios.....	52
Figura 16. Mensaje de error por credenciales erróneas o usuario no creado .....	52
Figura 17. Vista Dashboard de la plataforma web.....	53
Figura 18. Ventana informativa del ID del experimento y hora de inicio del experimento.....	54
Figura 19. Vista Experiments de la plataforma web.....	54
Figura 20. Mapa de ubicación de electrodos e identificación asociada obtenida de [28].....	54
Figura 21. Vista "Matriz" con mensaje indicativo de la hora de inicio de los estímulos con carácter indicativo de deletreo .....	55
Figura 22. Matriz con cambio de color en columna realizando el estímulo al paciente.....	55

Figura 23. Vista Statistics de la plataforma web .....	56
Figura 24. Vista Settings de la plataforma web .....	57
Figura 25. Visualización de mensajes en la vista de Settings.....	58
Figura 26. Vista Communication de la plataforma web .....	59
Figura 27. Subsección "Information to send" con conexión Local .....	60
Figura 28. Ejemplo de activación del bot con un chat previamente creado .....	60
Figura 29. Diagrama de apps para gestión de información en el backend .....	61
Figura 30. Interfaz de usuario de swagger para gestión de endpoints .....	62
Figura 31. Interfaz de acceso a plataforma de administración de usuarios .....	63
Figura 32. Vista principal de la plataforma de gestión de usuarios .....	63
Figura 33. Vista de gestión de usuarios .....	64
Figura 34. Vista de interfaz de adquisición de señales .....	65
Figura 35. Diagrama de relación entre tamaño de ventana y tamaño de desplazamiento en ventanas deslizantes .....	68
Figura 36. Modelo UML para la vista Login e interacción entre el paciente y supervisor	74
Figura 37. Modelo UML para la vista "Communication" entre la plataforma web y el dispositivo móvil.....	74
Figura 38. Modelo UML para la vista "Experiments" con respecto al generador de estímulos y el supervisor de los procedimientos .....	75
Figura 39. Modelo UML para la vista "Settings" bajo el rol de supervisor .....	75
Figura 40. Diagrama de Flujo general del sistema .....	76
Figura 41. Configuración base para ejecución de pruebas .....	80
Figura 42. Conexión de electrodos a voluntarios para ejecución de pruebas a) Voluntario A b) Voluntario B c) Voluntario C .....	81
Figura 43. Visualización de señales por canal en un conjunto de muestras, en cada uno de los electrodos en diferentes instantes .....	82
Figura 44. FFT del conjunto de canales de las señales en el dominio del tiempo en diferentes instantes .....	82



Figura 45. Grafica de calificación de época por voluntario.....	86
Figura 46. Grafica de precisión del clasificador por voluntario .....	86
Figura 47. Visualización de información respecto al desarrollo de un procedimiento en la vista Statistics .....	87
Figura 48. Comunicación de información desde la plataforma web al bot de Telegram con la información correspondiente .....	88
Figura 49. Arquitectura del sistema .....	95
Figura 50. Modelo Entidad-Relación de la base de datos.....	96
Figura 51. Tarjeta Cyton con especificación de pines por canal obtenida de [28] .....	97
Figura 52. Conexión de electrodos a la tarjeta Cyton obtenido de [28] .....	98
Figura 53. Conexión de fuente de alimentación a tarjeta Cyton obtenida de [28] .....	98
Figura 54. Indicación de pines de activación para tarjeta Cyton obtenida de [28] .....	99
Figura 55. Activación para funcionamiento dispositivo USB para comunicación con tarjeta Cyton obtenida de [28] .....	99
Figura 56. Vista de acceso tipo Login .....	101
Figura 57. Vista principal de plataforma web.....	102
Figura 58. Vista de Experimentos de la plataforma web .....	102
Figura 59. Especificación de botón para generación de ID de experimento .....	103
Figura 60. Alerta de ID y hora de inicio del experimento generado .....	104
Figura 61. Indicación de botón para abrir vista de matriz .....	104
Figura 62. Mensaje informativo en la vista matriz sobre hora de inicio .....	105
Figura 63. Parpadeo de filas y columnas en la vista de matriz .....	105
Figura 64. Visualización de cantidad de épocas por sección y secuencias.....	106
Figura 65. Visualización de cantidad de secuencias y porcentaje de precisión.....	107
Figura 66. Visualización del carácter detectada .....	107
Figura 67. Vista de configuración, seccion de informacion de paradigma.....	108

Figura 68. Vista de configuración, seccion de informacion general .....	109
Figura 69. Vista de configuración, seccion de procesamiento de señales .....	109
Figura 70. Vista de configuración, seccion de informacion de interfaz local .....	110
Figura 71. Vista de configuración, seccion de informacion de configuración de la matriz	111
Figura 72. Vista de comunicación con el dispositivo móvil con configuración general del bot	112
Figura 73. Envío local o integrado de información al dispositivo móvil .....	113
Figura 74. Visualización de estado de comunicación.....	113
Figura 75. Visualización de documentación de API en el backend.....	114
Figura 76. Plataforma de administración de usuarios.....	115
Figura 77. Administración de grupos de personas.....	116
Figura 78. Administración de modelos para configuración de bot.....	117
Figura 79. Administración de tokens de acceso .....	117
Figura 80. Administración de usuarios ya creados .....	118
Figura 81. Panel de modificación y actualización de usuarios .....	120
Figura 82. Panel de creación de usuarios.....	121
Figura 83. Busqueda de bot para creación de otros bots en Telegram .....	121
Figura 84. Comando para creación de nuevo bot .....	122
Figura 85. Comando para asignación de nuevo nombre al bot.....	122
Figura 86. Asignación de username del nuevo bot.....	122
Figura 87. Información de creación del nuevo bot .....	123
Figura 88. Interfaz de adquisición de señales con especificación de escritura del ID del experimento .....	124
Figura 89. Mensaje de experimento encontrado y fecha de inicio de pruebas .....	124
Figura 90. Mensaje informativo al usuario de experiación del ID del experimento .....	125

## 1. INTRODUCCIÓN

Con el avance tecnológico que se presenta actualmente, el presente proyecto relacionado en el área de las interfaces cerebro-ordenador (BCI), campo que posee una creciente aplicación dado el interés con la comunicación e interacción entre los seres humanos y las máquinas, se centra en el diseño e implementación de una interfaz BCI basada como una plataforma web que permite a los usuarios configurar los procedimientos ejecutados para deletreo de caracteres por medio del paradigma P300 ser personalizable para cada uno de ellos, de fácil uso, accesible desde cualquier dispositivo o ubicación e incluir comunicación con la aplicación móvil Telegram.

El sistema incluye algoritmos para la adquisición, filtrado, procesamiento, clasificación e identificación de las letras objetivo de deletreo por parte del paciente o voluntario del procedimiento, con configuraciones para garantizar la seguridad, almacenamiento y control de los datos realizados por cada usuario. La implementación de estos algoritmos con un elemento de machine learning permitirá aumentar la precisión de deletreo conforme las muestras obtenidas en cada procedimiento.

Este documento presenta los antecedentes a nivel nacional e internacional, con la respectiva relación de la problemática, contexto y justificación. Además, se relaciona cada una de las secciones y etapas del diseño, implementación y validación del sistema con los respectivos resultados para varios voluntarios en relación a la verificación de funcionamiento del sistema.

## **2. PLANTEAMIENTO DEL PROBLEMA**

### **2.1 Antecedentes y Estado del Arte**

#### **Nivel Internacional**

El propósito de una interfaz cerebro-ordenador (BCI) es poder detectar y analizar las características de las señales electroencefalográficas y comunicarlas en tiempo real utilizando modelos matemáticos a dispositivos de integración y análisis [1], los cuales servirán como base para procesos adicionales de control, visualización o procesamiento en múltiples áreas.

Varios referentes han creado diversos desarrollos basados en esta interfaz. Tal es el caso de las investigaciones generadas por algunos centros de investigación de ingeniería de IA y robótica en unión con el Ministerio de Educación de Shanghái en China [2] Estos resultados han permitido expresar un estándar unificado para el diseño y uso del potencial evocado P300 para actividades de deletreo, control de objetos o análisis de información de funciones cognitivas ligadas con acciones de toma de decisiones. Este estudio se enfocó en la descripción de las mejores técnicas, métodos y algoritmos que presenten alto nivel de precisión en los resultados cuando se utilizan en diversos requerimientos, permitiendo así establecer una base sobre las limitaciones y ventajas que puede tener la onda P300 en proyectos o investigaciones enfocados a dichas actividades.

La investigación presentada por VA Bulanov, AV Zajarov y EV Khivintseva [3] de Universidad Médica Estatal de Samara, se orienta a la determinación de la confiabilidad de un algoritmo basado en la transformada de Wavelet continua para la identificación de componentes P300, exponiendo los elementos matemáticos que constituyen dicho algoritmo. Inicialmente se indica las familias de señales Wavelet seleccionadas para la implementación de la transformada cuya aplicación se realiza por funciones contenidas en el software MATLAB. Partiendo de varios tipos y características de estímulos visuales propios del conjunto de datos registrados de señales EEG de acceso público, utilizados como fuente de información para el estudio, se identificaron los conjuntos de datos en donde se presentaba mayor intensidad del componente P300 respecto a la cantidad de épocas generadas para la

separación y aplicación del algoritmo diseñado. Los resultados expresaron la comparación en la precisión entre las diferentes familias de señales Wavelet con respecto a cada uno de los grupos de datos seleccionados, así como la del porcentaje de precisión general del algoritmo presentado. Esta investigación establece resultados aceptables [3] que pueden entregar los algoritmos basados en la transformada de Wavelet continua para la detección de componentes P300 en conjunto de datos procesados de señales EEG independiente de la plataformas o software en donde se implementen.

Otro proyecto desarrollado por Juan Manuel Ramírez-Cortes, Vicente Alarcon-Aquino, Gerardo Rosas-Cholula, Pilar Gomez-Gil y Jorge Escamilla-Ambrosio [4] para la detección de componentes P300 utilizando el algoritmo de clasificación *Adaptive Neuro Fuzzy* (ANFIS) y la transformada de Wavelet continua mediante la adquisición de señales EEG cuando varios sujetos son expuestos a estímulos visuales conforme a una secuencia de dispositivas en tiempo determinado, usando el componente electrónico para adquisición de señales de la empresa Emotiv de 14 electrodos. El sistema diseñado parte de la adquisición de señales con comunicación vía Bluetooth o puerto serial directo al computador que posteriormente genera el procesamiento, filtrado, extracción y clasificación con descomposición por frecuencias y bandas, lo que permitió observar e identificar los electrodos con mayor presencia de componentes P300 después de los estímulos establecidos, generando resultados sobre la viabilidad del uso del algoritmo con los electrodos específicos.

### **Nivel Nacional**

De igual forma en la Universidad El Bosque se realizó un proyecto de grado por los estudiantes Gendelfav Izquierdo y Luis Enrique Gómez de Ingeniería Electrónica[5] enfocado en integrar técnicas de adquisición y procesamiento de señales electroencefalográficas para interacción con una interfaz de usuario focalizados en el uso de las señales para la mejora de procesos atencionales y de concentración basándose en un proceso de integración entre los sistemas electrónicos de análisis de señales y *Machine Learning* (Aprendizaje de maquina) con teoría del desarrollo y salud del ser humano. Actualmente se presentan varios enfoques que incluyen interfaces BCI para la adquisición de señales EEG que facilitan la comunicación para personas con limitaciones motoras [6].

Esto se debe a que este tipo de personas pueden exhibir casos de críticos de depresión, soledad, ansiedad o conductas que afectan la salud del mismo en relación con las afecciones medicas que presentan. Diversos resultados han demostrado que la aparición de estos síntomas se relacionados con la disminución de las funciones cognitivas de secciones específicas del cerebro, que relaciona directamente con un mayor riesgo de presentar enfermedades neurodegenerativas [7].

Otro proyecto presentado por la universidad de los Andes por parte de L. Barbosa, C. Espinosa-Jovel, J. Vergara, M. Le Van Quyen, M. Valderrama [7], permite observar la generación de estímulos visuales y auditivos, con el uso de 19 electrodos bajo la configuración 10-20, con respecto al comportamiento de pacientes con epilepsia y como los componentes P300 se visualizan bajo condiciones de la enfermedad diferentes para cada paciente, entregando así una correlación de los datos para cada electrodo. De igual forma se observa una prueba de permutación para comparar la latencia de las ondas P300, seccionado por el tipo de estímulo para hacer la comparación con el análisis de varianza. [7] Este proyecto entrega información asociada a relación y comparación de características de las formas de onda con respecto a condiciones específicas, estableciendo una base para investigaciones posteriores que permitan comprender los casos médicos de estos pacientes.

La investigación presentada por el grupo de investigación conformado por: Eliana García Cossío, Carlos Fernández, María Eugenia Gaviria, Catalina Palacio, Liliana Alvarán y Robinson Alberto Torres Villa [8], se enfoca en el desarrollo y evaluación de una interfaz cerebro-computador basada en el potencial evocado P300, utilizada como herramienta de comunicación alternativa para personas con discapacidad motora. El estudio incluyó la participación de dos personas que presentaban desórdenes motore para lo cual, se utilizó una matriz de estimulación de 4x3 con imágenes de futbolistas contemporáneos. Los resultados obtenidos revelaron que los usuarios lograron controlar de forma adecuada la interfaz, alcanzando altos porcentajes de aciertos, entre en un 70% a 80%.

## 2.2 Descripción y Formulación del Problema

Conforme a los avances de la tecnología y su interacción del ser humano con esta, surge una amplia necesidad de desarrollar interfaces cerebro-computador (BCI) de fácil entendimiento, accesibles, con amplia retroalimentación y con alta flexibilidad para poder interactuar o comunicarse con otras tecnologías.

El paradigma P300 basado en la detección de respuestas cerebrales a estímulos visuales o auditivos inusuales a 300 milisegundos posteriores a su generación, es uno de los modelos con mayor aceptación para investigaciones de las interfaces BCI dada la facilidad de implementación, registro, adquisición y procesamiento, que a su vez son integradas con softwares específicos para poder ser funcionales para procedimientos o experimentos específicos. Sin embargo, a pesar de los múltiples avances se presentan las interfaces BCI presentan varias problemáticas a la hora de ser implementadas tales como:

1. **Eficiencia:** Algunas interfaces presentan difícil parametrización sobre todas las variables necesarias para la ejecución de los procedimientos de deletreo limitando la capacidad de usuarios o pacientes que puede utilizar dicha interfaz dados los requerimientos de las pruebas.
2. **Precisión:** La aplicación de modelos matemáticos específicos pueden generar sesgos sobre los resultados sobre el paradigma, estableciendo sesgos sobre los caracteres identificados.
3. **Adaptabilidad:** La ejecución de los procedimientos por partes de las interfaces no son versátiles para cada usuario o paciente que las use, generando disminución en la efectividad de la interacción del público con dicha tecnología.
4. **Operabilidad:** Las interfaces no son intuitivas y de difícil aprendizaje para los pacientes o especialistas, produciendo retardos en la aplicabilidad de los procedimientos.
5. **Seguridad:** La manipulación de los resultados, señales, y datos de procesamiento de cada uno de los experimentos son de manejo local (se guardan en el mismo

dispositivo que se opera) lo que implica riesgos de seguridad y acceso a dicha información

El presente proyecto desarrolló e implementó una interfaz BCI que utiliza métodos no invasivos para capturar señales del usuario, ejecutar un procesamiento y clasificación de las mismas para generar comunicación e interacción con otros usuarios de grupo social utilizando dispositivos móviles. La implementación implica clasificación, cuantificación e inducción selectiva de ondas P300 por medio de estímulos visuales, lo que permite el reconocimiento de caracteres para escritura de mensajes complejos. Las plataformas web y las aplicaciones móviles como Telegram permiten a múltiples usuarios visualizar instantáneamente la recopilación de datos de múltiples procedimientos con características y controles similares, ajustando los parámetros y condiciones según sea necesario durante la experiencia o necesidades de cada usuario.

Para la adquisición de señales se utiliza una tarjeta por integración con electrodos no invasivos de acuerdo con las características necesarias para la comunicación con las tecnologías de implementación correspondientes. El procesamiento, transmisión y visualización de la información se realiza a través de una serie de procedimientos que garantizan la autenticidad y seguridad de los datos tanto de los participantes como del público antes, durante y después del evento.

### **2.3 Justificación**

El presente proyecto de grado busca diseñar e implementar una interfaz de comunicación BCI utilizando el potencial evocado P300 con un sistema de comunicación TCP/IP hacia dispositivos móviles o plataformas web. Para esto se implementarán procesos no invasivos de adquisición de señales electroencefalográficas, que permitan a los pacientes expresar sus necesidades, opiniones o requerimientos mediante actividades de deletreo usando técnicas BCI, ya que estas tecnologías actualmente ofrecen diversos métodos para facilitar dicha funcionalidad, incluyendo además una tarjeta por integración para adquisición, filtrado y amplificación de las señales.



El procesamiento de señales electroencefalográficas con una interfaz BCI compatible con múltiples protocolos, plataformas y tecnologías de comunicación permitirá tener un mayor acceso, fácil integración, menor cantidad de recursos y bajo costo al ser integrado con sistemas que se encuentran actualmente en el mercado dado el acceso a la interfaz sin requerimientos de instalación o guardado de información local, así como la ejecución de los experimentos bajo una configuración personalizada para cada paciente que generará mayor precisión en las mediciones en un menor tiempo de adaptabilidad.

Construir la interfaz bajo un modelo de escalabilidad vertical, permite a los usuarios controlar el precio conforme a la cantidad de datos almacenados y utilizados para los diversos procedimientos. Así comparado con respecto a los softwares del mercado cuyo valor solamente incluye la adquisición sin mantenimiento, presenta una considerable disminución de costos para los pacientes que la necesiten.

Dependiendo del estado del paciente y la técnica de adquisición de las señales EEG generadas bajo la estimulación visual, las ondas P300 producidas en el usuario que mediante algoritmos se realizar el procesamiento, filtrado y clasificación de las muestras para ser identificada la presencia o no de estas ondas por cada una de las épocas generadas. Además de integrar la interfaz de adquisición y la plataforma web con dispositivos móviles por medio de tecnologías de acceso libre y con implementación personalizada de acuerdo con los requerimientos del proyecto.

## **2.4 Objetivos**

### **Objetivo General**

Desarrollar una interfaz BCI para la comunicación asíncrona en dispositivos móviles usando métodos no invasivos de adquisición de señales electroencefalográficas por medio del paradigma P300.

### Objetivos Específicos

1. Desarrollar un algoritmo de adquisición, procesamiento, caracterización y clasificación de datos por medio de técnicas de *machine learning*, usando el paradigma P300
2. Desarrollar un sistema de comunicación TCP/IP para el envío de mensajes de texto a interfaces de mensajería o red social.
3. Implementar la plataforma web con los procesos de autorización de personal y envío de información

### 2.5 Alcance y Limitaciones del Proyecto

Para este proyecto se realizará el proceso de adquisición, amplificación y filtrado de señales electroencefalográficas mediante una tarjeta por integración ya definida que permita obtener el potencial evocado P300. Se implementará un sistema de comunicación entre el software de la interfaz BCI con dispositivos móviles, además de diseñar e implementar una plataforma web para la visualización y control de la información obtenida de los experimentos utilizando el lenguaje de programación *Python*.

El presente proyecto tiene como limitaciones:

1. Usar métodos no invasivos de adquisición de señales electroencefalográficas
2. El sistema deberá ser desarrollo en 3 semestres académicos universitarios
3. El análisis y caracterización de las señales electroencefalográficas debe ser basada en la onda P300.
4. Los pacientes que participen en los procedimientos que posean limitaciones motoras deben poseer un consentimiento escrito y aprobado por dicho paciente. El proceso de la gestión y aprobación de la participación de los pacientes en los experimentos estará a cargo del codirector del proyecto, perteneciente al área de psicología de la Universidad

### 3. MARCO DE REFERENCIA

#### 3.1 Marco Teórico o Conceptual

##### Transformada de wavelet

Representa un método analítico que implementa la capacidad de identificar eventos de frecuencia localizados temporalmente. A diferencia de la transformada de Fourier de corto tiempo (STFT), que mantiene una resolución constante durante el análisis, WT ajusta dinámicamente su resolución durante todo el proceso.[9], [10]

La diferencia fundamental entre STFT y WT radica en la adaptabilidad de WT. La WT, mediante el uso de funciones de descenso, puede ajustar su resolución para capturar tanto detalles temporales como espectros de frecuencia de eventos específicos. [9] Esta función  $\psi(t)$  debe satisfacer ciertas propiedades matemáticas para que sea útil en el análisis de señales [10]. La transformada wavelet se define de la siguiente manera:

$$W_f(s, \tau) = \int f(t) \psi_{(s, \tau)}^*(t) dt \quad (A1)$$

Una wavelet es un conjunto de funciones derivadas de una función original llamada “wavelet principal”  $\psi(t)$ . Estas wavelets se obtienen mediante dos procesos básicos: desplazamiento y escalamiento de la wavelet original. La traducción implica mover la wavelet original a lo largo del eje del tiempo, mientras que el escalado implica estirar o comprimir la wavelet original a lo largo del mismo eje [10]. Esta wavelet principal se define como:

$$\psi_{(s, \tau)}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t - \tau}{s}\right) \quad (A2)$$

##### Transformada de wavelet discreta

La Transformada de Wavelet Discreta (DWT) es una técnica de análisis de señales que permite descomponer una señal en diferentes componentes de frecuencia y tiempo. A diferencia de la Transformada de Fourier, que utiliza funciones sinusoidales como base, la

DWT utiliza funciones wavelet, que son funciones de onda más cortas y localizadas en el tiempo.[9]

La DWT se obtiene al discretizar los parámetros de desplazamiento y escalamiento dentro de la Transformada de Wavelet Continua. Los valores utilizados para esto suelen ser  $a = 2^{-j}$  y  $b = k2^{-j}$  donde  $a$  es el escalamiento,  $b$  es el desplazamiento y  $j, k$  son valores enteros.[9], [10]  
La función wavelet madre en la DWT queda de la siguiente manera:

$$\psi_{(s,\tau)}(t) = 2^{\frac{j}{2}} \psi(2^j t - k); j, k \in \mathbb{Z}$$

La DWT se define como:

$$DWT_{j,k} = 2^{\frac{j}{2}} \int_{-\infty}^{\infty} x(t) \psi(2^j t - k) dt$$

Donde  $x(t)$  es una señal discreta. DWT Proporciona una forma eficiente de representar y analizar señales en términos de sus características de frecuencia y escala, siendo especialmente útil en la codificación de señales y la compresión de datos, el procesamiento de señales de audio, el análisis de series temporales y la eliminación de ruido de datos.[10]

### **Familias de señales de wavelet**

Las funciones wavelet pueden considerarse una familia de funciones que tienen buena localidad tanto en frecuencia como en tiempo. [11] En la teoría de wavelets, basada en la teoría de conjuntos, puede representar la frecuencia y el tiempo de la señal.[12]

Para el análisis de señales, la transformada Wavelet proporciona una representación de la señal en frecuencia y tiempo.[12] La transformada wavelet es útil para analizar señales no estacionarias porque ofrece resolución de frecuencia variable y temporal, a diferencia de la transformada de Fourier.[13]

La transformada wavelet se utiliza para reducir el ruido de la señal del campo eléctrico generada por un rayo. También se ha utilizado para estudiar el tráfico fractal en redes de comunicación. Además [13], la transformada wavelet se utiliza en el procesamiento de señales EEG para extraer características y detectar y analizar el potencial P300.

### **Ondas EEG y bandas asociadas**

En neurociencia, el electroencefalograma (EEG) es una herramienta importante para registrar la actividad eléctrica en el cerebro. La dinámica cerebral genera señales eléctricas que podemos capturar mediante electrónica, acoplamiento de impedancia preciso y sensores especiales de ganancia de bajo ruido. Estas señales, también conocidas como electroencefalogramas o EEG, muestran la actividad de las células nerviosas en una variedad de patrones de intensidad, intensidad y frecuencia. [14]

El EEG está diseñado para decodificar la actividad eléctrica del cerebro y su variación en respuesta a diversos factores. El diagnóstico y tratamiento de trastornos neurológicos como la epilepsia y otros dependen de la interacción entre la actividad cerebral y las señales EEG. [14] En el espectro de las señales EEG, la intensidad típica varía entre 0 y 200 microvoltios ( $\mu V$ ), presentando frecuencias que abarcan hasta 50 Hz, y ocasionalmente, alcanzando los 100 Hz en casos particulares. En individuos sanos, emergen distintas bandas de frecuencia en el EEG: Alpha, Beta, Theta, Delta y Gamma. Cada una de estas bandas está relacionada con diferentes estados de actividad cerebral y rasgos específicos. [14], [15]

- **Delta:** Las ondas Delta, con su rango de frecuencia de 0.2 a 4 Hz, predominan durante los estados de sueño y relajación profundos. Son indicativas de procesos regenerativos y reparadores del cerebro, y su presencia es más evidente en las primeras etapas del sueño. [14], [15]
- **Theta:** Las ondas Theta, que oscilan entre 4 y 8 Hz, emergen en estados de relajación profunda, meditación y momentos de transición entre el sueño y la vigilia. Estas ondas permiten que el cerebro explore conexiones más flexibles entre diferentes áreas, posibilitando la generación de ideas y la asociación libre. [14], [15]

- **Alpha:** La banda Alpha, con su rango de frecuencia de 8 a 14 Hz, se manifiesta cuando estamos en un estado de relajación tranquila pero alerta. También pueden indicar un estado de calma y concentración, como cuando nos sumergimos en una meditación profunda o una actividad que requiere atención sostenida.[14], [15]
- **Beta:** Las ondas Beta, que abarcan de 12 a 30 Hz, son un reflejo de la actividad mental activa y la concentración. Cuando estamos despiertos y comprometidos en tareas cognitivas, las ondas Beta se vuelven prominentes. Diferentes sub-rangos de las ondas Beta pueden estar asociados a diferentes niveles de activación y enfoque. [14], [15]
- **Gamma:** La banda Gamma, con su rango de frecuencia de 30 a 90 Hz, juega un papel crucial en la integración de información neuronal de diferentes áreas cerebrales. Las ondas Gamma reflejan una actividad cerebral sincronizada y coordinada en redes neurales más amplias.[14], [15]

### **Paradigma P300**

El potencial P300, es un potencial cerebral llamado así por sus características de polaridad y latencia, Aproximadamente 300 ms después de la presentación del estímulo, se produce una desviación positiva en el registro del EEG [16]. El estímulo que provoca el P300 debe cumplir ciertos criterios, como ser inusual y/o sorprendente para el sujeto, además de ser relevante para la tarea o experiencia.

La relevancia del estímulo está directamente relacionada con la amplitud del P300. En otras palabras, la amplitud del P300 aumentará con el estímulo más relevante. Además, la probabilidad del estímulo está inversamente ligada a la amplitud [16]. Se espera que, después de la presentación del estímulo, esta señal alcance su punto máximo en aproximadamente 300 milisegundos (ms) [7] Sin embargo, varios estudios han demostrado que P3 puede manifestarse en un rango de tiempo de 250 a 650 (ms) después de la exposición al estímulo. [7], [17]

La Figura F1 muestra una variedad de variantes del paradigma Oddball, cada una de las cuales tiene una variación en la forma en que se muestran los estímulos. Una de ellas muestra

un solo estímulo poco común [18]. En todos los casos, los sujetos deben mantener una atención especial y responder mental o físicamente al estímulo poco común. La amplitud ( $\mu\text{V}$ ) se calcula como la diferencia entre el voltaje promedio previo al estímulo y el punto más alto del pico positivo en la forma de onda del ERP dentro de un intervalo de tiempo específico [18]. La latencia (ms) se refiere al tiempo transcurrido desde la presentación del estímulo hasta el momento de máxima amplitud positiva dentro de un intervalo de tiempo determinado. La distribución del P300 en el cuero cabelludo se caracteriza por el cambio en la amplitud sobre los electrodos centrales (Fz, Cz, Pz), con una tendencia típica de aumento desde los electrodos frontales hacia los parietales.[19]

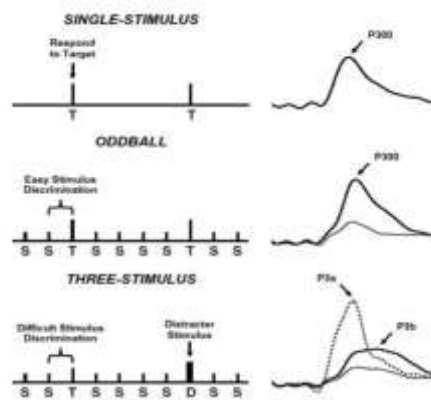


Figura F1. Protocolos conocidos para generar respuestas P300. Imagen extraída de [18]

## Interfaz BCI

Una interfaz cerebro-computadora (BCI) es una tecnología que permite convertir la actividad neural registrada desde el cerebro en instrucciones que se pueden usar para controlar una computadora u otro dispositivo externo [20]. El término "BCI basado en el potencial P300" se refiere a un sistema de interfaz cerebro-computadora que utiliza la señal de control del potencial P300. El potencial P300 es una señal con un pico de amplitud que se observa en el EEG alrededor de 300 milisegundos después de la presentación de un estímulo auditivo o visual poco común [2], [20], [21]. En este enfoque, se presenta al usuario una serie de estímulos, y solo algunos de ellos están relacionados con la intención del usuario. Los estímulos de interés, al ser poco frecuentes y mezclarse con otros estímulos más comunes,

provocan la aparición del potencial P300 en la actividad cerebral del usuario, principalmente en las áreas central y parietal del cerebro[22]

Por otro lado, los pacientes con discapacidad suelen utilizar sistemas de deletreo BCI. Estos sistemas requieren que el usuario combine caracteres, números y símbolos para formar palabras. El sistema de deletreo BCI analiza la señal de EEG del usuario para encontrar el carácter deseado en el paradigma de deletreo. Luego, crea un comando de control que el usuario puede usar. Estos sistemas de deletreo se clasifican según su enfoque en las señales del EEG. Los paradigmas incluyen deletreo híbrido, P300, potenciales evocados visuales de estado estable (SSVEP) e imaginación motora (MI). [21].

### **Algoritmo clasificador OneRule**

El clasificador OneRule o algoritmo OneR, un método de clasificación sencillo y rápido se utiliza para predecir la clase de una variable categórica basándose en una sola regla [3]. Su objetivo principal es crear reglas de clasificación que se adapten mejor a la mayoría de los datos de entrenamiento. En el aprendizaje automático supervisado, este método se utiliza en función de la frecuencia de cada valor de la variable predictora y la clase a la que pertenece. [3], [23] El proceso de clasificación utilizando el algoritmo OneRule en el contexto del aprendizaje automático implica un enfoque de "máquina de aprendizaje supervisado".[23] Primero, se preparan los datos, que incluye limpiar y transformar los datos de entrenamiento para que sean adecuados para el análisis. Luego, el algoritmo OneRule selecciona una característica principal en función de su capacidad para predecir la variable objetivo.

A continuación, se describe el proceso general del algoritmo OneR:

- **Preparación de datos:** Antes de aplicar el algoritmo OneR, es fundamental realizar una limpieza y transformación de los datos de entrenamiento. Esto implica manejar valores faltantes, codificar variables categóricas en números, y asegurarse de que los datos estén listos para su análisis.[24]
- **Selección del atributo principal:** El algoritmo OneR identifica la característica o atributo que mejor se ajusta a la variable objetivo. Para hacerlo, calcula el error



asociado a cada atributo. El atributo que resulta en el menor error al crear una regla se selecciona como el atributo principal. En otras palabras, se busca el atributo cuyos valores proporcionen la mejor predicción de la clase. [3], [25]

- **Generación de reglas:** Una vez que se ha seleccionado el atributo principal, el algoritmo OneR crea reglas de clasificación para cada valor posible de ese atributo [25]. Cada regla asocia un valor del atributo con una clase particular. Por ejemplo, si el atributo principal es "Edad" y se está prediciendo si alguien compra o no un producto, el algoritmo podría crear reglas como "Si la Edad es menor de 30 años, entonces Clase = Compra" y "Si la Edad es igual o mayor de 30 años, entonces Clase = No Compra".
- **Evaluación del rendimiento:** Una vez que se han generado estas reglas, el rendimiento del algoritmo se evalúa utilizando datos experimentales, que no se utilizaron durante el entrenamiento [25]. Se evalúa cuántas instancias se clasifican correctamente y cuántas se clasifican incorrectamente utilizando las reglas creadas

### 3.2 Marco Legal o Normativo

Con base en la normativa vigente en Colombia, se establecen leyes que supervisan la protección de datos personales en el ámbito de la investigación científica y tecnológica. La Ley 1581 de 2012 y sus decretos reglamentarios establecen las directrices que aseguran la integridad de los datos personales y biomédicos, regulando su recolección, almacenamiento y uso [26]. Estas normativas adquieren especial importancia en proyectos como la interfaz cerebro-computadora (BCI), al garantizar la privacidad y los derechos de los participantes en la investigación, pruebas, procedimientos o experimentos en los que participen [26].

Igualmente, es fundamental considerar la normativa vigente en el ámbito de la investigación en el área de salud. La Resolución 8430 de 1993 del Ministerio de Salud y Protección Social establece los cimientos científicos, técnicos y administrativos para la ejecución de investigaciones de salud en Colombia [27]. Esta resolución establece las pautas éticas que rigen la investigación en seres humanos y clasifica los niveles de riesgo inherentes a los estudios [27]. Estas regulaciones éticas y científicas concurren para asegurar la calidad y pertinencia de las investigaciones en este ámbito expuesto.

## **4. DESARROLLO DEL PROYECTO DE GRADO**

### **4.1 Requerimientos**

#### **4.1.1 Requerimiento General**

El sistema deberá recibir señales EEG de un paciente que recibe entradas visuales de una pantalla, identificar las letras a la vez para componer un mensaje, transmitir esa información a una plataforma web y desde allí a la aplicación Telegram de dispositivo móvil, por medio de un Bot.

#### **4.1.2 Requerimientos funcionales**

**RF01:** El sistema permitirá la visualización de señales ECG de forma gráfica, configuración autónoma y personalizada para cada usuario para el procesamiento de la información y comunicación con el aplicativo móvil en la interfaz gráfica de la plataforma Web.

**RF02:** La plataforma web debe ser accesible por medio de un computador y/o celular con acceso a internet.

**RF03:** El código fuente del sistema estará almacenado en un repositorio de proveedores en la nube de código cerrado con control de versiones.

**RF04:** El sistema permitirá al usuario cargar el mensaje compuesto por letras individuales no mayor a 20 caracteres junto con la información de conexión y comunicación del destinatario del mensaje.

**RF05:** El sistema llevará un registro de los experimentos realizados, así como la información del paciente que realizó cada experimento.

**RF06:** La plataforma web permitirá el ingreso de distintos usuarios bajo reconocimiento de usuarios registrados y persistencia de información.

**RF07:** El sistema será multiplataforma, es decir funcionará en los distintos tipos de sistemas operativos desde donde se ejecute.

**RF08:** El sistema incluirá algoritmos adecuados para el ventaneo, procesamiento e identificación de las muestras de acuerdo con los requerimientos del paradigma P300.

**RF09:** El sistema debe permitir modificar los parámetros de ventaneo, visualización, interacción, clasificación, comunicación y adquisición necesarios para el funcionamiento de la plataforma o interfaz.

**RF10:** El sistema partirá desde el proceso de adquisición de señales electroencefalográficas realizado por una tarjeta por integración.

**RF11:** El sistema permitirá al usuario modificar la información del *bot* con el que se generará la comunicación.

#### 4.1.3 Requerimientos no funcionales

**RNF01:** El ingreso al sistema será restringido bajo usuarios y contraseñas cifradas, utilizando método de autenticación en token de acceso.

**RNF02:** El sistema contará con una única versión de funcionamiento.

**RNF03:** La tarjeta por integración para la adquisición de señales electroencefalográficas deberá cumplir con la normatividad asociada para la medición por métodos no intrusivos en pacientes.

**RNF04:** El sistema contará con un sistema de mensajes informativos al usuario

#### 4.1.4 Requerimientos de calidad

**RC01:** El sistema deberá cumplir con el plan de pruebas establecido que permita garantizar la seguridad de la información, comunicación entre los dispositivos y funcionamiento de los procesos de detección de información basada en el paradigma P300.

**RC02:** El sistema deberá obtener mínimo un 60% en la precisión de los procesados por el clasificador

**RC03:** El sistema garantizará el envío y transmisión de datos entre la plataforma web y Telegram como aplicación del dispositivo móvil.

#### 4.1.5 Requerimientos de restricción

**RR01:** Los experimentos se deben realizar con pacientes adultos con una edad mínima de 18 años.

**RR02:** El paradigma base para identificación de señales es P300.

**RR03:** Inclusión de tarjeta por integración para adquisición de señales EEG, el hardware utilizado será proporcionado por el laboratorio de psicología experimental de la universidad el Bosque.

**RR04:** Los estímulos para la detección de las ondas P300 deben ser visuales.

**RR05:** Se usará Telegram como aplicación móvil para realizar la comunicación entre la plataforma web y el dispositivo móvil.

**RR06:** La tarjeta por integración a utilizar para la adquisición de señales EEG será la tarjeta Cyton de OpenBCI.

**RR07:** Se usarán de 4 a 8 electrodos que se puedan integrar a la tarjeta de adquisición de señales.

**RR08:** Los lenguajes de programación a utilizar serán *Python*, *CSS*, *HTML*, *JavaScript* y como motor de base de datos se usará *PostgreSQL*.

**RR09:** Los experimentos y pruebas se realizarán con pacientes sanos.

**RR10:** El sistema tendrá el protocolo TCP/IP para realizar la comunicación entre la plataforma web y la aplicación móvil.

**RR11:** La matriz de deletreo puede ser configurable compuesta de letras y números, contará con fuente tipo ROBOTO con un tamaño entre 12 y 16 puntos en letra mayúscula.

**RR12:** El sistema contará con una guía de uso (manual de usuario) con las especificaciones necesarias para el manejo de la herramienta.

## **4.2 Metodología del Diseño**

La metodología utilizada para el desarrollo del presente proyecto se centra en tres secciones específicas: diseño, implementación y validación, que describen el orden de ejecución y los componentes relacionados al desarrollo del proyecto.

### **4.2.1 Diseño**

#### **4.2.1.1 Diseño General**

Para establecer el diagrama de caja negra se tienen en cuenta la interacción entre los usuarios y la interfaz BCI, además de la interacción entre la adquisición de señales, los estímulos visuales y la clasificación de señales, esta interacción se puede observar en la figura 1.

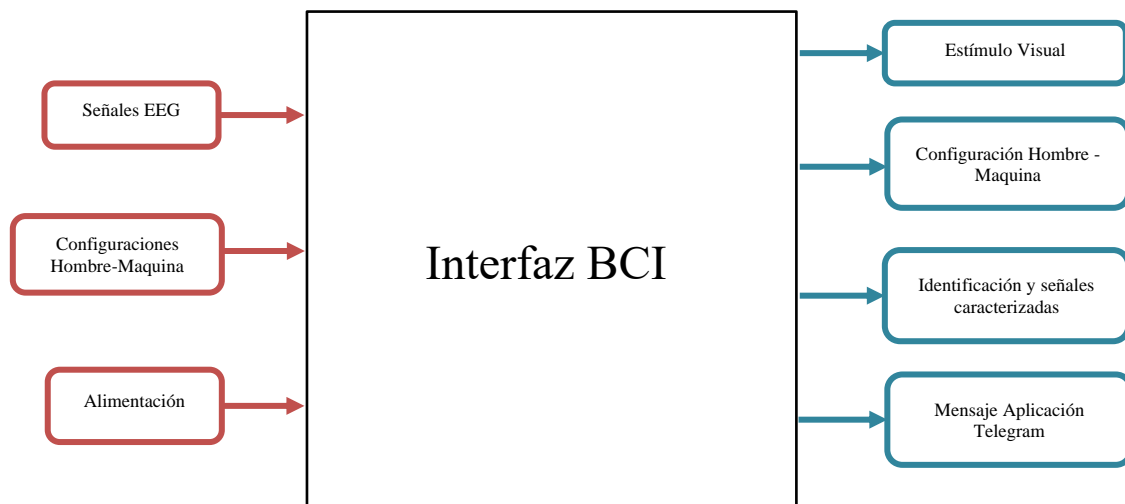


Figura 1. Diagrama de caja negra

#### 4.2.1.2 Diseño Funcional

En la figura 2 se presenta el diagrama funcional de acuerdo con el diagrama general del proyecto presentado en la figura 1, en el cual cada una de las secciones a desarrollar dentro del desarrollo de la interfaz

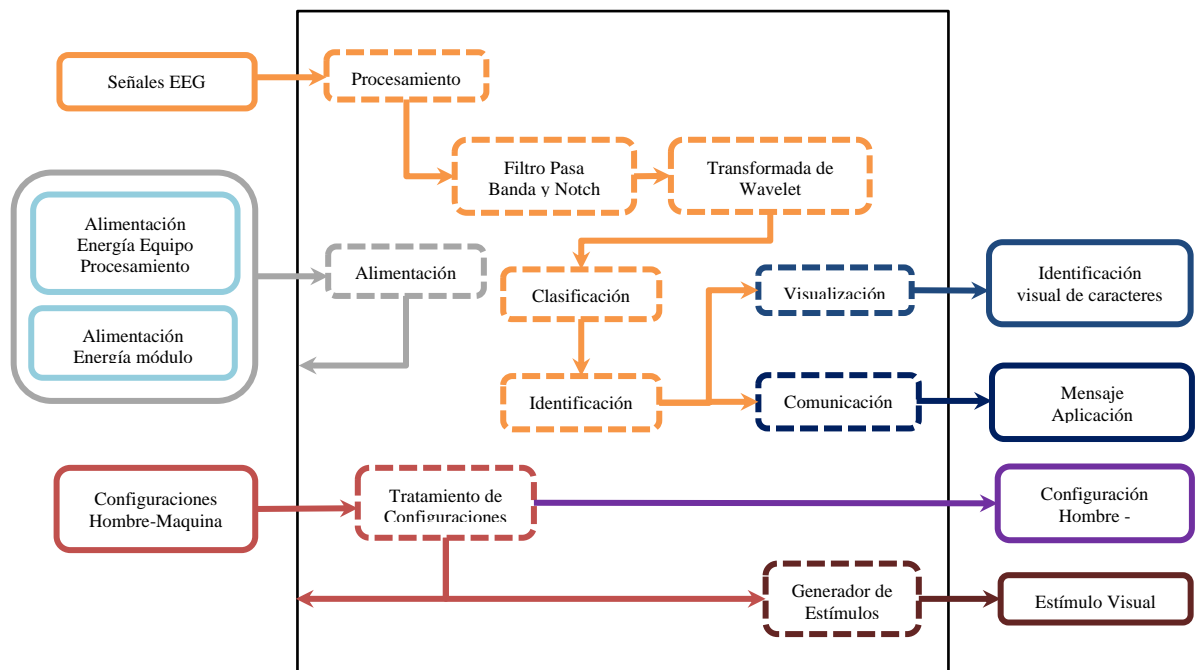


Figura 2. Diagrama funcional basado en el diagrama de caja negra

El diagrama funcional presenta varios colores para la identificación de los flujos entre entradas, salidas y módulos internos de la interfaz BCI. La descripción de cada flujo se define a continuación:

- **Color Naranja:** Flujo de tratamiento desde que se realiza la adquisición de señales por medio de la tarjeta hasta la identificación del carácter de acuerdo con el procesamiento generado.
- **Color Rojo:** Flujo que presenta las configuraciones realizadas por el usuario acorde a los experimentos. Las configuraciones son aplicables a la mayoría de módulos de acuerdo con los requerimientos y necesidades de los mismos.
- **Color Café:** Flujo de la generación de estímulos visuales basado en las configuraciones del usuario
- **Color Azul:** Flujo asociado a la visualización y comunicación con los dispositivos (Aplicaciones y pagina web) respecto a la información procesada por el sistema.
- **Color Morado:** Flujo correspondiente a la visualización y almacenamiento de las configuraciones realizadas por el usuario.
- **Color Gris:** Flujo que presenta la alimentación del sistema y que es utilizado en todos los módulos de la interfaz.

#### 4.2.1.3 Diseño Detallado

Para cada uno de los módulos que componen la interfaz BCI expresados en el diagrama funcional de la figura 2 a continuación, se describe la composición de cada uno de ellos con respecto a su actividad y relación con los demás.

##### **Procesamiento**

En este módulo se realiza la adecuación y separación de los datos obtenidos de la tarjeta de adquisición de señales. Por medio de la interfaz de adquisición, permitirá generar la conexión de la tarjeta con el computador utilizando el puerto COM del computador realizando así la lectura de los datos de acuerdo con la frecuencia de muestreo de la tarjeta de adquisición y

la cantidad de canales conectados. El diagrama interno de este componente se expone en la figura 3.

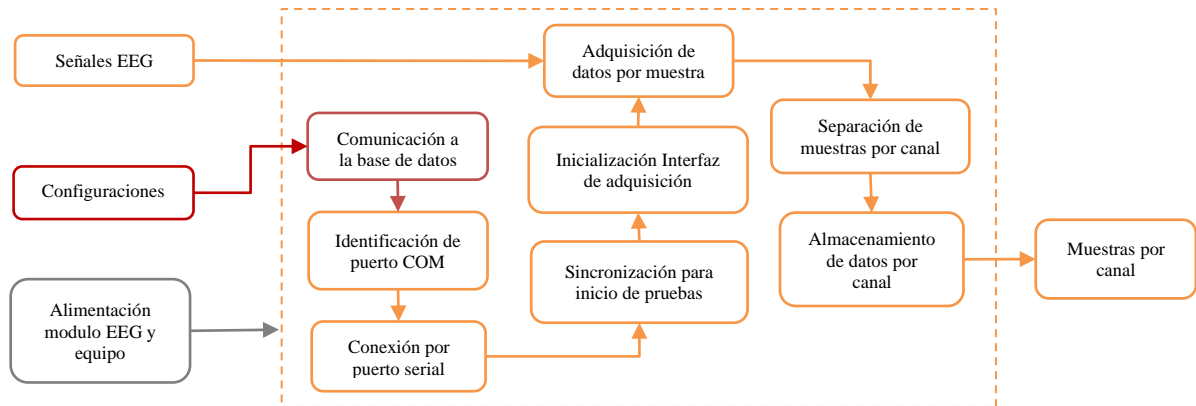


Figura 3. Diagrama interno para módulo de procesamiento

### Filtro Pasa Banda y Notch

El filtro pasa banda es diseñado en las frecuencias de corte de 0.3 Hz a 50 Hz dado que las familias de ondas a analizar que pueden contener componentes P300 se encuentran en dicho rango. De igual forma se implementa un filtro Notch de 60 Hz para eliminar la frecuencia de operación de la red alimentación de la mayoría de sistemas electrónicos. Cabe resaltar que el orden del filtro pasa banda y la cantidad de muestras a analizar de las ventanas deslizantes para aplicación de los filtrados son configuradas por el usuario. La relación de este procedimiento se presenta en la figura 4.

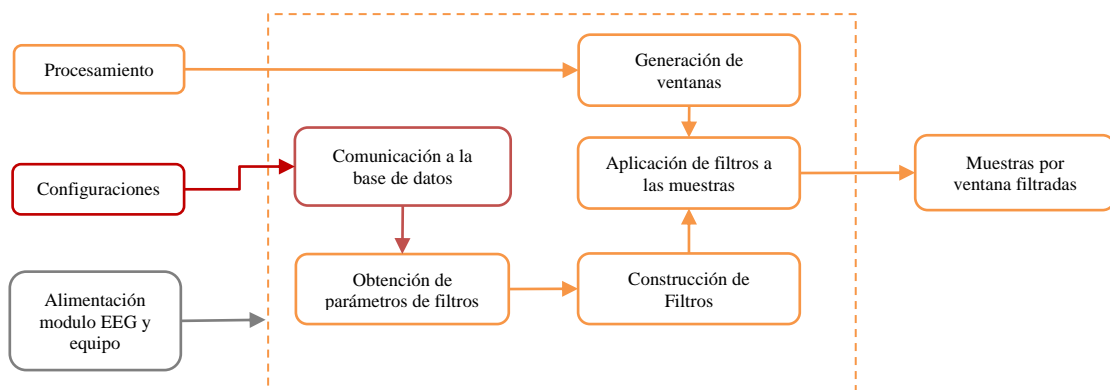


Figura 4. Diagrama interno para módulo de filtro pasa banda y notch

### Transformada de Wavelet

La transformada de Wavelet permite descomponer la señal de cada canal en diferentes componentes de frecuencia y tiempo, dado que la onda P300 puede presentarse en intervalos no regulares de tiempo, de acuerdo con los estímulos presentados. Al incluir una familia wavelet nos permite obtener una representación de la señal con patrones en escalas, lo que facilita la detección de la onda P300 por medio de otros algoritmos. Para esta sección se establecen los coeficientes para cada una de las ventanas analizadas y su respectiva reconstrucción de la señal por transformada inversa, que permita resaltar los componentes P300 en cada canal. La figura 5 presenta el diagrama de esta interacción.

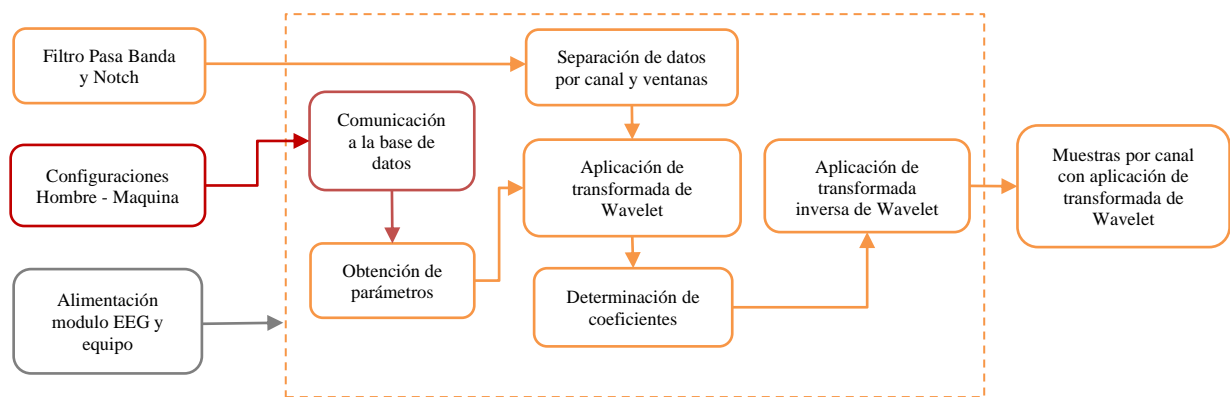


Figura 5. Diagrama interno para módulo de transformada de wavelet

## Clasificación

Para el módulo de clasificación se integran algoritmos de categorización que permiten identificar en espacios y muestras la presencia de componentes P300. El correcto funcionamiento de este módulo depende directamente del tratamiento de las señales para el realce de características que afectan directamente la precisión del clasificador, esta información se guarda en la base de datos con respecto al ID del experimento para la respectiva visualización en la matriz de estímulos. El diagrama que representa la funcionalidad de este módulo se presenta en la figura 6.



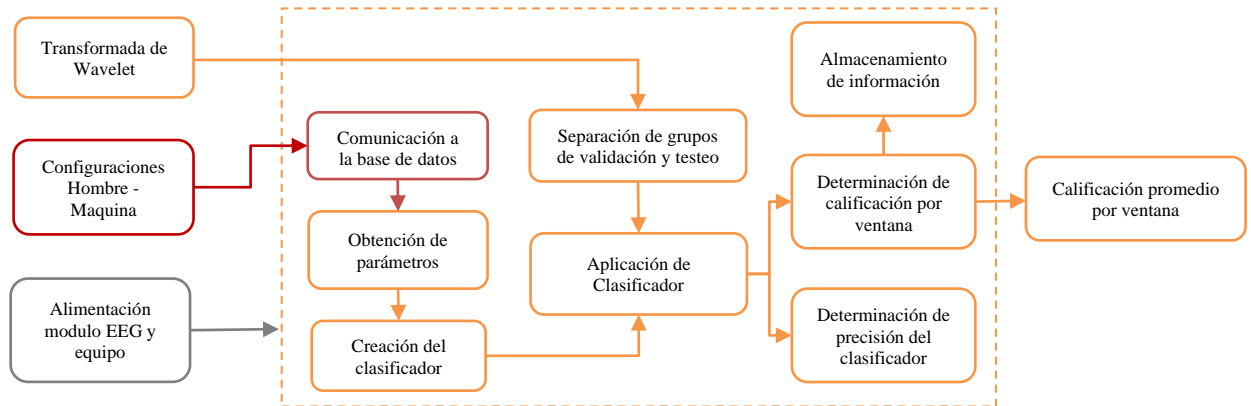


Figura 6. Diagrama interno para módulo de clasificación

## Identificación

En la vista de matriz de estímulos en la cual se generan los parpadeos y las respectivas épocas, el sistema realiza la consulta a la base de datos para obtener los datos de clasificador con respecto a cada época y secuencia, con el fin de correlacionarlos con el orden de las épocas y visualizar la letra identificada. Esta información es ejecutada para cada una de las letras que el usuario desea deletrear. El procedimiento previo se puede observar en la figura 7.

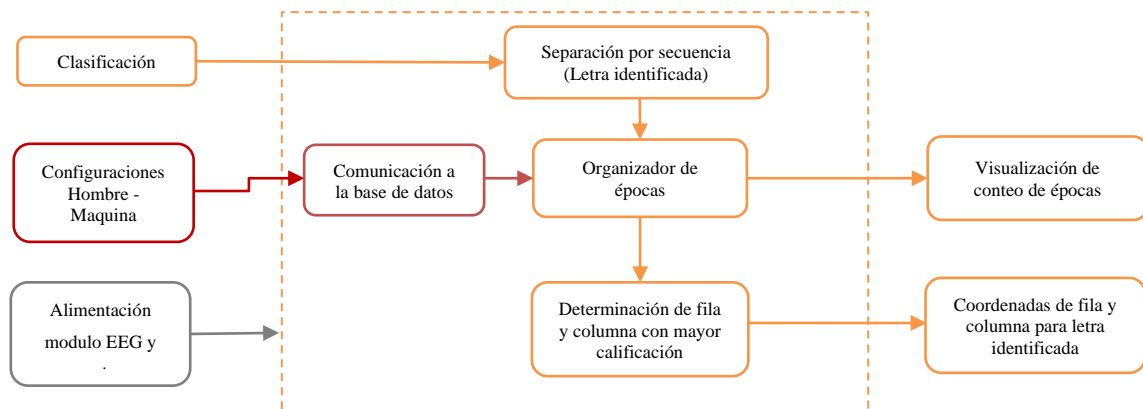


Figura 7. Diagrama interno para módulo de identificación

## Visualización

El módulo de visualización presenta la retroalimentación de los algoritmos de clasificación e identificación de las ondas con respecto a la matriz de estímulos. De igual forma se presentan los componentes necesarios para que el usuario pueda interactuar con la interfaz,

realizar comunicación con el dispositivo y visualización de información característica de las señales. La composición interna de este módulo se puede observar en la figura 8.

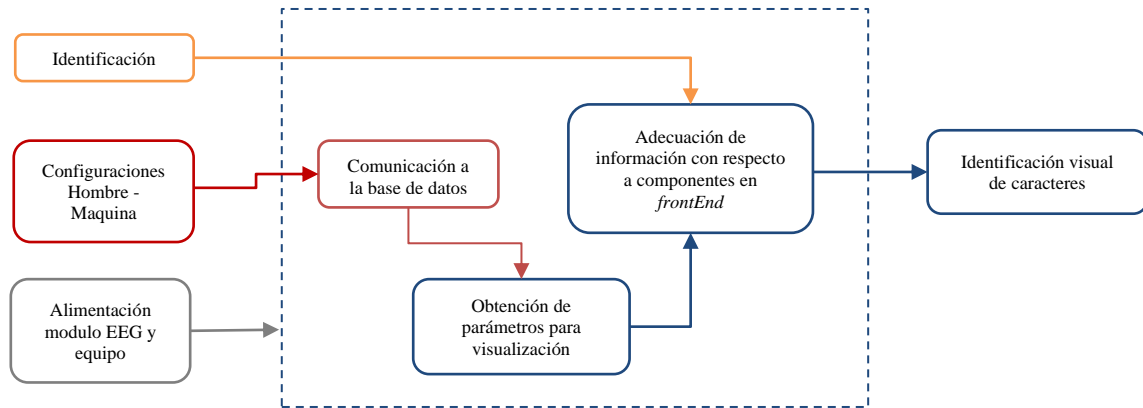


Figura 8. Diagrama interno para módulo de visualización

## Comunicación

La comunicación con el dispositivo móvil desde la plataforma web respecto a la identificación de los caracteres identificados pertenece a este módulo, donde se integran los algoritmos necesarios para la conexión con la API de Telegram, la conexión al ID del *bot* y su respectivo envío de información. En la figura 9 se puede observar las secciones que componen este módulo.

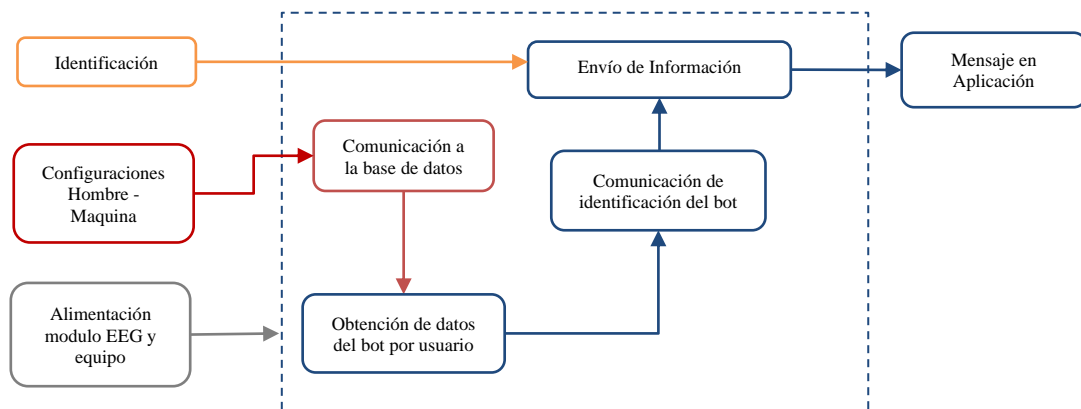


Figura 9. Diagrama interno para módulo de comunicación

### Tratamiento de configuraciones

Las configuraciones realizadas por el usuario en la plataforma web son almacenadas por usuario en la base de datos del sistema, estas configuraciones son aplicables a todos los módulos, ya que los parámetros ingresados por el usuario son factores indispensables en el tratamiento de señales, generación de estímulos e identificación de caracteres. El procedimiento para el almacenamiento, visualización y comunicación con los otros módulos se puede observar en la figura 10.

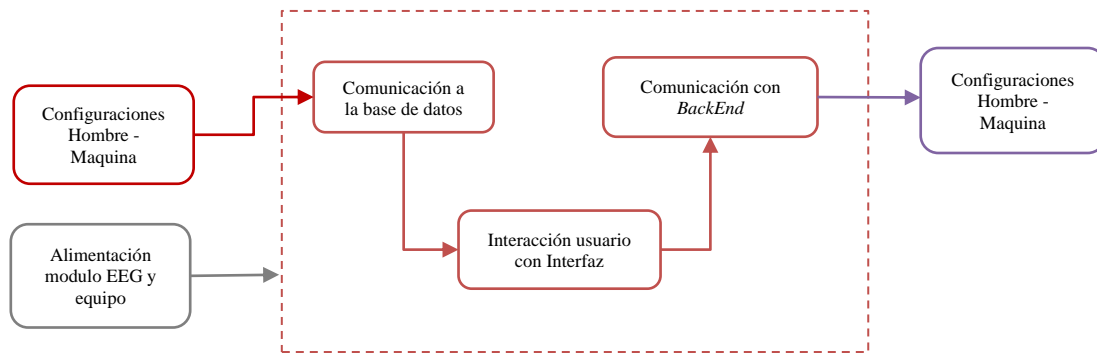


Figura 10. Diagrama interno para módulo tratamiento de configuraciones

### Generador de estímulos

En este módulo presenta la generación de estímulos en intervalos de tiempo, orden aleatorio y secuencial en la matriz de estímulos. Varios de los parámetros que controlan dicha generación son configuraciones realizadas por el usuario en el módulo correspondiente de la plataforma web y se adecuan para que la generación sea acorde con necesidades del usuario. El diagrama que permite observar el procedimiento se observa en la figura 11.

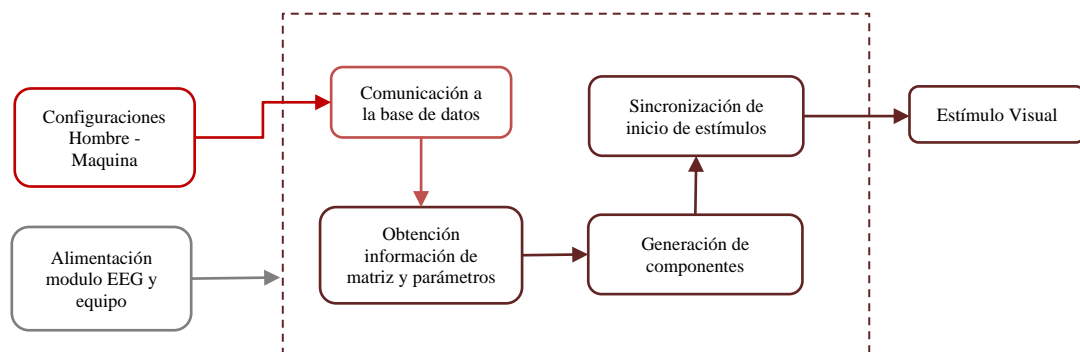


Figura 11. Diagrama interno para módulo generador de estímulos

## Alimentación

La alimentación de los sistemas de procesamiento se realiza de acuerdo a los requerimientos de los dispositivos en donde se vaya a acceder a la plataforma web o a la aplicación de Telegram, los servidores en donde se despliegue la aplicación y los dispositivos adicionales que se conecten a los equipos de procesamiento o visualización de información.

### 4.2.1.4 Selección de componentes, tecnologías y técnicas

Para el diseño e implementación de la plataforma web y la comunicación con los dispositivos correspondientes de debe realizar la selección de cada uno de los elementos a integrar, este proceso se presenta en las siguientes secciones.

## Formato de bits para procesamiento

El formato de bits para los datos de salida de la placa de adquisición se describe como se expresa en la figura 12:

- Byte 1: Contador de paquetes
- Byte 2: Número de muestra
- Bytes 3-5: valor de datos para el canal EEG 1
- Bytes 6-8: valor de datos para el canal EEG 2
- Bytes 9-11: valor de datos para el canal EEG 3
- Bytes 12-14: valor de datos para el canal EEG 4
- Bytes 15-17: valor de datos para el canal EEG 5
- Bytes 18-20: valor de datos para el canal EEG 6
- Bytes 21-23: valor de datos para el canal EEG 6
- Bytes 24-26: valor de datos para el canal EEG 8
- Bytes 27-28: valor de datos para el canal X del acelerómetro
- Bytes 29-30: valor de datos para el canal Y del acelerómetro
- Bytes 31-32: valor de datos para el canal Z del acelerómetro

Figura 12. Formato Binario para tarjeta Cyton de OpenBCI obtenida de [16]

De acuerdo con el requerimiento **RF10** de la sección 4.1. del presente documento, la tabla 1 expresa la relación de las características por formato de bits:

Tabla 1. Características por formato de datos para selección para selección de técnica.

Característica	16 bits	24 bits	32 bits
Incluye cantidad el envío de todos los canales	No	Si	Si
Permite compresión de datos	Si	Si	Si
Incluye datos por impedancia de canal	No	No	Si
Establecen marcas de tiempo por muestra	Si	Si	Si
Discriminación por muestras	Si	Si	Si

De acuerdo con las necesidades y requerimientos del sistema, no se necesita conocer la posición u orientación de los electrodos de la tarjeta, por lo cual el formato elegido es el de 24 bits, el cual no integra datos bytes de posición, pero integra toda la información de los canales y discrimina los datos por canal.

### Configuración de posición de los electrodos en la cabeza del paciente

Existen distintos tipos de configuraciones estándar para los electrodos y de acuerdo con el requerimiento **RR07** de la sección 4.1. del presente documento, la tabla 2 describe las características con respecto a la configuración de electrodos:

Tabla 2. Características de posición de electrodos para selección de modelo

Característica	10-5	10-10	10-20
Cantidad máxima de electrodos	4	10	16
Resolución espacial	Alta	Muy alta	Alta
Área de cobertura	Todo el cuero cabelludo	Todo el cuero cabelludo	Todo el cuero cabelludo
Profundidad de registro	Superficial	Superficial	Superficial y profunda
Dificultad para colocación	Moderada	Moderada	Difícil
Sensibilidad al ruido	Media	Media	Alta
Costo	Bajo	Medio	Alto

Los criterios de selección dependen la cantidad de electrodos a utilizar y el uso de los electrodos no implique realizar intervenciones al paciente, bajo esta información la configuración seleccionada es 10-10 dada la capacidad de soportar 8 electrodos de captura de señales, electrodo de referencia y electrodo de tierra, además de la implementación superficial del paciente y con facilidad de aplicación.

### Librerías de implementación

Dado el requerimiento **RR08** de la sección 4.1. del presente documento, el cual integra los lenguajes de programación de funcionamiento para la interfaz y procesamiento, la tabla 3 especifica la relación entre las características y las librerías más utilizadas para adquisición de señales EEG.

Tabla 3. Características por librería base de uso para selección de procedimiento

Característica	pyOpenBCI	MNE-Python	Brainflow	NeuroKit
Soporte de hardware	Varios dispositivos EEG	Varios dispositivos EEG	Varios dispositivos EEG	Varios dispositivos EEG
Procesamiento de señales avanzado	Incluido	Incluido	No incluido	Incluido
Soporte de señales	EEG, MEG, ECG	EEG, MEG, ECG.	EEG, EMG, ECG.	EEG, EMG, ECG.
Procesamiento de señales	No incluido	Incluido	Incluido	Incluido
Visualización de señales	incluido	Incluido	Incluido	Incluido
Documentación	Amplia y detallada	Amplia y detallada	Limitada	Amplia y detallada
Compatibilidad con sistemas operativos	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux

Basado en la información de la tabla anterior *pyOpenBCI* es la mejor opción si se utiliza la *Cyton Biosensing Board*, ya que incluye la configuración y secciones de código especiales para la tarjeta a utilizar. Además, *pyOpenBCI* cuenta con documentación completa y en constante actualización, lo que facilita el proceso de adquisición de datos, preprocesamiento y análisis. Finalmente, la librería cuenta con soporte activo lo cual favorece el uso de la librería.

### Técnicas para la detección de la onda P300

Basado en el requerimiento **RF08** de la sección 4.1. del presente documento, existen varios métodos matemáticos y estadísticos para la detección de la onda P300, la cual es la base para realizar la clasificación de las señales. Las características por cada tipo de técnica se especifican en la tabla 4:

Tabla 4. Características por técnica para detección de la onda

Característica	PCA	ADF	ACC	WT
Poseen alta precisión en la detección de la onda	Si	Condicionado	Si	Limitado
Integra sensibilidad para detectar pequeños cambios	Media	Condicionado	Condicionado	Condicionado
Elimina la variabilidad de las señales	Si	Limitado	No	No
Permite la eliminación de los artefactos	No	Si	Limitado	Si
Velocidad de Procesamiento de las señales	Media Condicionada	Bajo	Media Condicionada	Alta
Sensible a la calidad de los datos	Si	Si	Si	Si

De acuerdo con la información relacionada, la técnica con mejores condiciones para eliminar los artefactos con una alta velocidad de procesamiento, condiciones necesarias para la comunicación y visualización de letras identificadas en la plataforma web es la técnica por transformada de wavelet

### Receptor de comunicaciones en Telegram

Al realizar la comunicación con Telegram existen varios tipos de entornos en donde se puede realizar esta comunicación, la cuales tienes configuraciones y características específicas, basándose en el requerimiento **RR05** de la sección 4.1. del presente documento, en la tabla 5 se describe las características por cada tipo de entorno que integra la aplicación:

Tabla 5. Características por tipo de receptor para selección de modelo

Característica	Chat	Bot	Grupo
Presenta identificador único	Si	Si	Si
Permite comunicación bidireccional entre los servicios	No	Si	Si
Permite creacion autoservicio por el usuario	SI	Si	Si
Posee integracion con API Telegram	No	Si	No
Integra envío de mensajes a contactos	Si	No	Si
Integra secuencia de comandos para comunicación	No	Si	No

Debido a la cantidad de características que la aplicación puede utilizar para procesos de comunicación autónoma, asincrónica, comunicación bidireccional y auto gestionable por cada uno de los usuarios es el entorno del bot. Estos criterios se asocian directamente a los requerimientos de comunicación entre la plataforma web y la aplicación de Telegram.

### Técnica de clasificación

Bajo los requerimientos **RF08**, **RF09** y **RC02** de la sección 4.1. del presente documento, la tabla 6 establece la comparación entre las distintas técnicas de clasificación y sus respectivos criterios de selección.

Tabla 6. Características por técnica de clasificación para selección

Característica	Promedio de Épocas	Filtrado Espacial	One Rule	ICA	ERP
Procesamiento de múltiples canales	No	Si	Si	Limitado	Si
Precisión de clasificación	Relativa	Alta	Alta	Media	Media
Complejidad de implementación	Baja	Alta	Media	Media	Media
Velocidad en procesamiento	Alta	Media	Alta	Media	Alta
Preprocesamiento requerido	No	Si	Si	Si	No
Necesidad de ventaneo de muestras	No	Si	Si	No	Si
Incluye validación cruzada	No	Si	Si	Dependiente	Si

Dado que el sistema requiere comunicación en tiempo real, que permita generar el análisis, identificación y visualización del carácter generado. La técnica de clasificación debe ser precisa, con alta velocidad de procesamiento y que permita incluir múltiples canales de acuerdo con los requerimientos del usuario. Bajo estas características la técnica que cumple con estos criterios es el clasificador *One Rule*.

### Tipo de Filtro

El tipo de filtro digital a implementar durante el procesamiento de la señal de acuerdo con la frecuencia descrita por el usuario es descrito en la tabla 7, teniendo en cuenta el requerimiento **RF08**.

Tabla 7. Características por tipo de filtro digital

Característica	Bessel	Butterworth	Chebyshev	Elíptico
Tipo de Respuesta	Lineal	Plana	Rizado	Rizado
Presencia de Distorsión	Baja	Baja	Moderada	Alta
Velocidad de transición	Lenta	Moderada	Rápida	Rápida
Aplicable en Python	Si	Si	Si	Si
Facilidad de Diseño	Moderada	Fácil	Moderada	Fácil

Ya que el filtrado de la señal no debe generar distorsión en las señales debido a la amplitud de las señales y que la velocidad de procesamiento del filtro debe ser óptima para el



procesamiento de las ventanas por la cantidad de canales, el tipo de filtro elegido es el Butterworth

### ***Framework para frontend***

Actualmente se presentan varios *framework* que se basan *JavaScript* y permiten integración con *HTML* y *CSS*, en la tabla 8 se relaciona la selección del *framework* para *frontend* de acuerdo con el requerimiento **RR08**.

Tabla 8. Características por framework basado en *JavaScript* para *frontend*

Característica	Next.Js	Angular	Vue.Js	Ember.Js
Escalabilidad	Alta	Alta	Media	Media
Velocidad de procesamiento	Alta	Alta	Media	Media
Capacidad de seguimiento de errores	Alta	Baja	Media	Baja
Soporte de funcionamiento	Alta	Alta	Media	Media
Tipado estricto	Medio	Alta	Alta	Alta
Facilidad implementación de arquitectura	Medio	Alta	Alta	Alta

Teniendo en cuenta que la aplicación debe ser escalable de fácil, que posea herramientas adecuadas para el seguimiento de errores y no posea tipado estricto para la creación de clases de comunicación el *framework* seleccionado es Next.Js

### ***Framework para backend***

De acuerdo con el requerimiento **RR08** es necesario realizar la selección del *framework* que se base en *Python* para la inclusión en el proyecto. La tabla 9 presente la relación de características con respecto a cada uno de los *frameworks*.

Tabla 9. Características por framework basado en *Python* para *backend*

Característica	Django	FastAPI	Flask
Escalabilidad	Alta	Media	Media
Velocidad de procesamiento	Alta	Media	Media
Procesos automáticos de validación de usuarios	Si	No	Bajo Integración
Soporte de funcionamiento	Alto	Alto	Medio
Plataforma de administración de usuarios	Si	No	No
Seguridad	Alta	Media	Baja

La velocidad de procesamiento es uno de los factores importante debido a la cantidad de peticiones que se realizaran por usuario y por secuencia, además de contener plataforma de administración de usuarios para aumento de la seguridad de datos y comunicaciones. Bajo estos criterios el *framework* seleccionado es Django.

#### 4.2.1.5 Diseño de Interfaz de usuario

##### Diseño de mockups

Como parte del diseño de la plataforma web, se establecen inicialmente los mockups de las principales ventanas sobre las cuales se generará la integración de la información, visualización y configuración de los experimentos. Dichos diseños se expresan en las figuras 13 a la 16.



Figura 13. Mockups para la interfaz (a) *Dashboard* (b) *Experiments* (c) *Settings* (d) *Communication*

### 4.2.2 Implementación

La construcción de las diferentes plataformas de acuerdo con el diagrama de funcional de la figura 2. Se divide en tres secciones integrales que permiten el cumplimiento de los diferentes requerimientos y funcionalidad del proyecto a continuación, se describen cada una de estas secciones:

#### 4.2.2.1 FrontEnd

La interfaz gráfica de la plataforma web es construida en el framework Next.JS el cual es basado en React para la construcción de aplicaciones e interfaces web. La gran escalabilidad, velocidad de carga y experiencia de usuario permiten que la visualización de los caracteres se pueda realizar sin modificar o remplazar los componentes ya generados. La interacción del usuario con la plataforma web se realiza por medio de vistas, cada una con las características necesarias para generar la comunicación y envío de información. Estas vistas se describen a continuación.

#### Vista *Login*

En esta vista el usuario realiza el proceso de autenticación con un *username* y contraseña, esta información previamente es ingresada por el administrador de usuarios en la plataforma correspondiente, esto permitirá al usuario acceder a la plataforma. La figura 14 permite observar la construcción de esta vista.



Figura 14. Vista principal Login de plataforma web

Al ser enviada los datos ingresados, se realiza una solicitud API REST al *Backend* para generar el Token únicamente si las credenciales ingresadas son correctas y previamente el usuario ha sido registrado. De acuerdo a lo anterior, la vista puede generar dos posibles estados, como se puede observar el siguiente diagrama de la figura 15.



Figura 15. Diagrama de flujo de la validación de credenciales y autenticación de usuarios

Para el estado de las credenciales erróneas o el usuario no esté creado previamente, el sistema generara el mensaje de error visualizado en la figura 16.

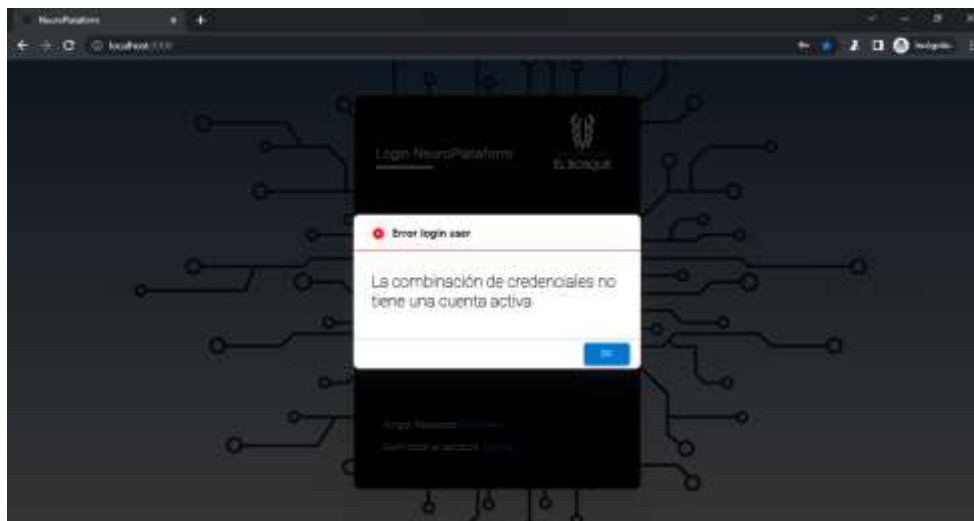


Figura 16. Mensaje de error por credenciales erróneas o usuario no creado

### ***Vista Dashboard***

Esta vista permite visualizar información relacionada con el proyecto, título del mismo y la funcionalidad del mismo. Además, es posible al visualizar imágenes o datos conforme las actualizaciones del sistema o sus relacionados, lo anterior se puede visualizar en la figura 17.

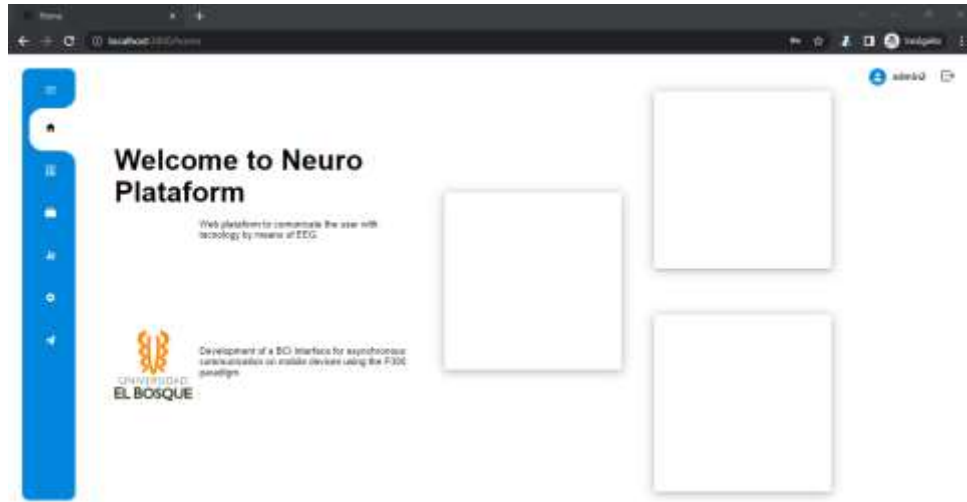


Figura 17. Vista *Dashboard* de la plataforma web

### ***Vista Experiments***

Esta vista permite generar los ID del experimento conforme el usuario registrado. Dado que los sistemas de adquisición y generación de estímulos requieren una sincronización de inicio, la generación del ID expondrá la hora de inicio, el cual es 4 minutos después del almacenamiento de la información en la base de datos como se puede apreciar en la figura 18. Con lo anterior el botón de apertura de la matriz de estímulos estará habilitado para iniciar con el experimento como se puede apreciar en la parte superior de la figura 19. Finalmente, esta vista presenta la imagen de la configuración de ubicación de los electrodos 10-10 en la parte central de la figura 20, la cual establece una guía al usuario sobre la posición de los mismos en el paciente y una breve visualización en la sección derecha de la figura 19, acerca de la trazabilidad de experimentos realizados por el usuario autenticado.

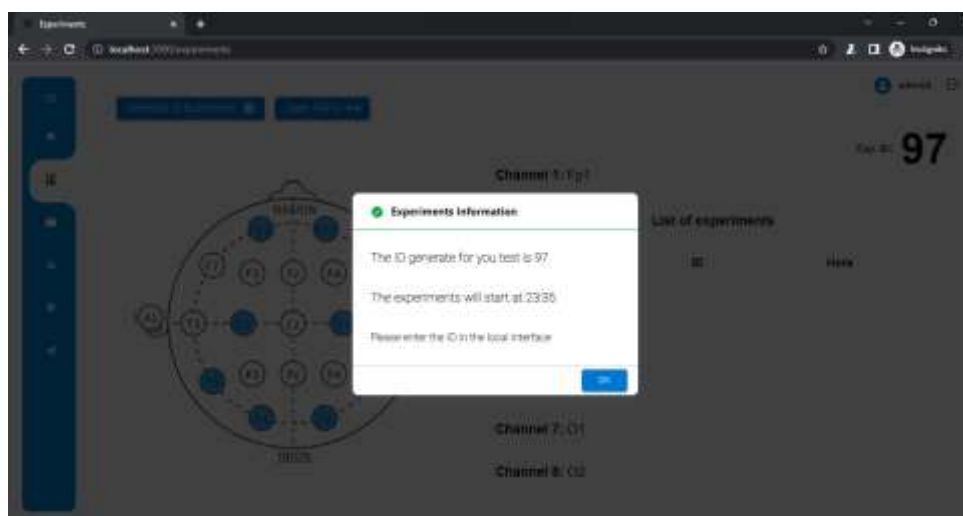


Figura 18. Ventana informativa del ID del experimento y hora de inicio del experimento

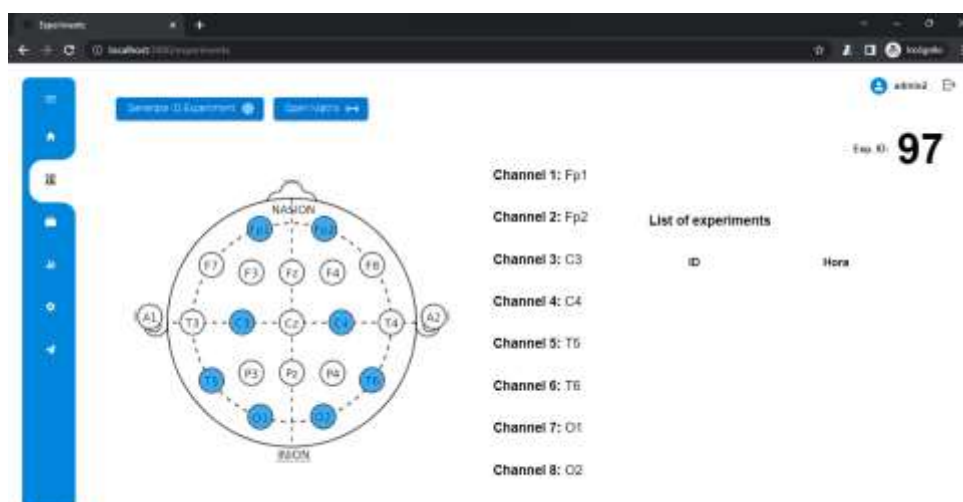
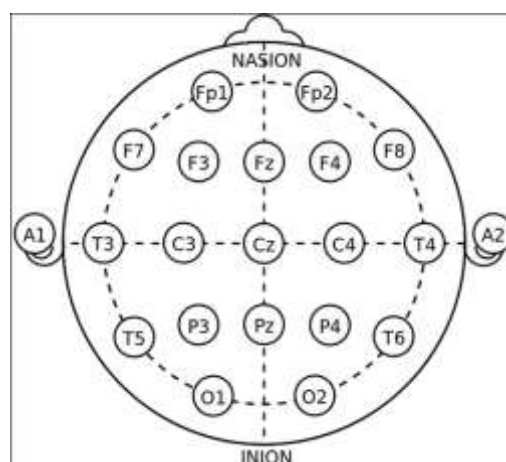
Figura 19. Vista *Experiments* de la plataforma web

Figura 20. Mapa de ubicación de electrodos e identificación asociada obtenida de [28]

### Vista Matrix

Para esta vista permite visualizar la matriz 6 x 6 con los caracteres configurados en la vista de Settings la cual realiza la consulta a la base de datos con el ID del usuario autenticado, que modificara su color con respecto al parpadeo (estímulo visual) de acuerdo con el tiempo dado por el usuario. La figura 21 presenta el mensaje informativo sobre la hora de inicio de la prueba con respecto al ID del experimento generado previamente, el cual se presenta al momento de ingresar a la vista. Por otro lado, la figura 22 presenta el cambio de color cuando se realiza un parpadeo de la vista.

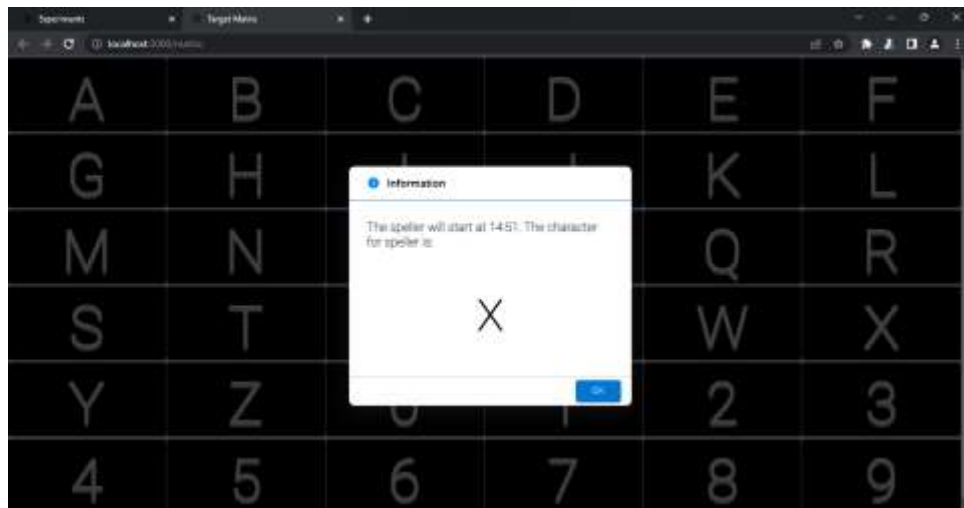


Figura 21. Vista "Matriz" con mensaje indicativo de la hora de inicio de los estímulos con carácter indicativo de deletreo



Figura 22. Matriz con cambio de color en columna realizando el estímulo al paciente.

Dependiendo de la cantidad de caracteres es generado las horas de inicio de estímulos para cada letra, además durante el proceso de entrenamiento del paciente, se especifica el carácter a deletrear como se puede observar en la figura 21.

### Vista Statistics

La presente vista permite visualizar el conteo la cantidad de épocas por cada una de las filas y columnas respecto a la cantidad de caracteres o secuencias ejecutado por la matriz, dicha información se visualiza únicamente cuando la matriz de estímulos esté activa. El diseño de la vista se puede apreciar en la figura 23.

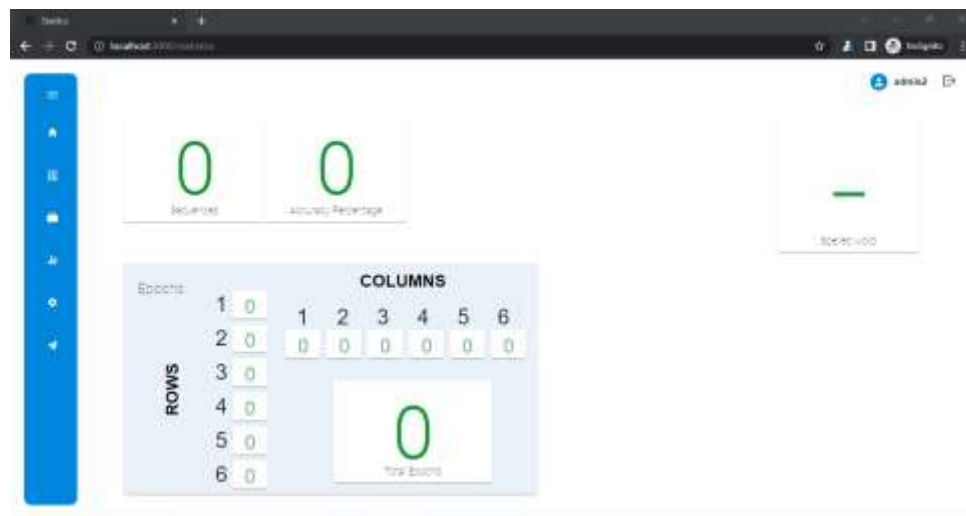


Figura 23. Vista *Statistics* de la plataforma web

### Vista Settings

La presente vista permite realizar la configuración de todos y cada uno de los parámetros integrados a la adquisición, filtrado, clasificación, identificación, visualización y comunicación que utiliza la plataforma web y la interfaz de adquisición de señales. Dicha información es parametrizada para cada usuario y almacenada en la base de datos, para consulta y uso en las secciones que los requieran. La figura 24 presenta la vista general de la vista, con su respectiva información base.



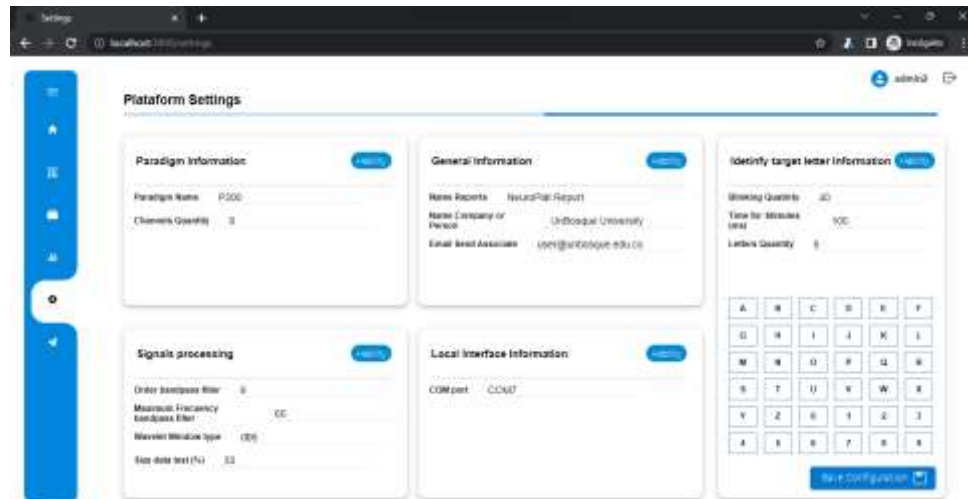


Figura 24. Vista Settings de la plataforma web

La vista presenta 5 subsecciones cada una de ellas relacionada una sección específica de la plataforma, cada una de ellas con los parámetros de configuración se relacionan a continuación:

- *Paradigm Information*: Indica al usuario el tipo paradigma a utilizar para deletreo (Componentes P300), además permite configurar el número de canales que utilizara la tarjeta para la adquisición de señales.
- *General Information*: Permite al usuario configurar el nombre asociado a los reportes entregados por la plataforma, así como la información general del usuario y entidad que utiliza la plataforma:
- *Signals Processing*: Proporciona al usuario realizar la configuración del orden y frecuencia de corte del filtro pasa banda, es de recalcar que la frecuencia de corte máxima depende directamente de la frecuencia de muestreo que utilice la tarjeta de acuerdo con el teorema de Nyquist. Además, proporciona selección del tipo de familia wavelet que utilizara la transformada de wavelet, así como el porcentaje de muestras a utilizar por el clasificador para testeo.
- *Local Interface Information*: Entrega la información de configuración del puerto COM en el cual se encontrar conectada la tarjeta de adquisición.
- *Identify target letter information*: El usuario permite realizar la configuración de la cantidad de parpadeos, tiempo entre parpadeos y la cantidad de letras a deletrear,

además se presenta cada uno de las casillas para la visualización de la matriz de estímulos el cual el usuario puede ingresar hasta 4 caracteres por casilla.

Finalmente, el sistema genera mensajes informativos en la parte superior derecha para comunicar al usuario estado específicos de la interacción con la plataforma, en la figura 25 se observan los mensajes expuestos por la plataforma.

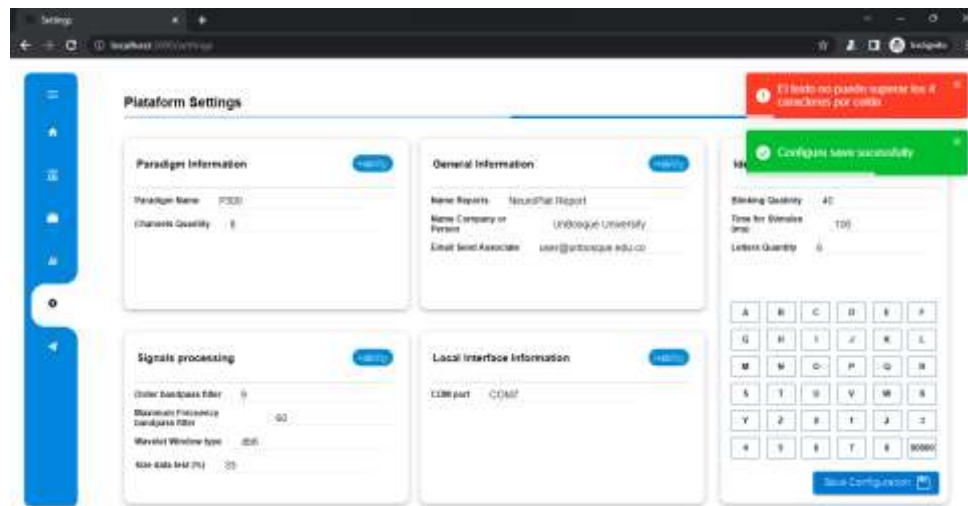


Figura 25. Visualización de mensajes en la vista de *Settings*

### **Vista *Communication***

Esta vista genera la comunicación de la plataforma web con *bot* del dispositivo móvil. Dicho procedimiento se realiza bajo la conexión de la API de Telegram desarrollada en el backend de la plataforma. La figura 26 presenta la vista general de la vista de comunicación.

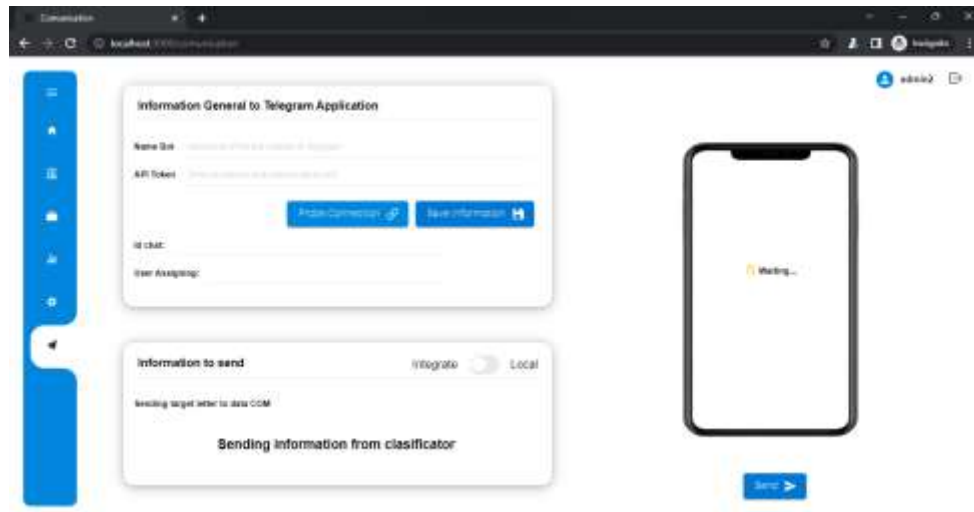


Figura 26. Vista *Communication* de la plataforma web

Esta vista presenta dos secciones de interacción con el usuario, estas se describen de la siguiente forma:

- *Information General to Telegram Application*: Para generar la conexión de la plataforma con el bot, para lo cual el usuario debe ingresar el nombre del *bot* y el API Token que entrega la aplicación de Telegram al momento de realizar la creación del *bot*. Con la información anterior con el interactuar con el botón “*Probe Connection*”, la vista enviara la información al backend que utiliza la API de Telegram por medio del protocolo TCP/IP, al indicarle una URL con la IP de conexión al servidor (La documentación de esta API se especifica en el anexo G del presente documento) para obtener el ID del chat y el propietario que genero el *bot*, verificando la correcta comunicación con la API para el envío de información, esta información es visualizada en los componentes inferiores de la sección. Finalmente, con los datos de conexión es posible guardar los resultados en la base de datos del sistema para uso posterior del usuario.
- *Information to send*: Esta sección realiza la configuración de la información que se enviara al *bot*, la conexión base es “*Integrate*” en donde el sistema al detectar información del carácter identificado por la vista de matriz será enviado inmediatamente al *bot*, sin embargo, es posible cambiarlo por botón a “*Local*” para

envío de información escrita por el usuario. La modificación de esta subsección en conexión local se visualiza en la figura 27.

Figura 27. Subsección "Information to send" con conexión Local

Para la realización de la conexión con el *bot* de Telegram es necesario realizar el inicio del mismo por medio del comando */start*, esto permitirá activar la comunicación del *chat* por medio de la API correspondiente. La figura 28 visualiza un ejemplo de activación con un *bot* creado por medio de la aplicación utilizando la versión de escritorio de la misma.

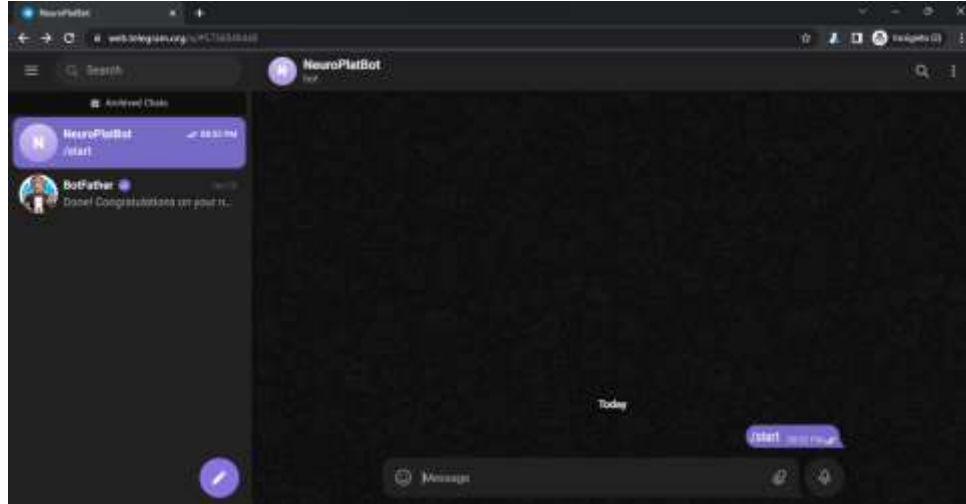


Figura 28. Ejemplo de activación del *bot* con un *chat* previamente creado

Dado que el usuario puede crear bot personalizados e integrarlo a la plataforma web para su correspondiente comunicación, el anexo D indica el procedimiento para ejecutar dicha acción.

#### 4.2.2.2 BackEnd

El backend es construido bajo el *framework* Django Rest Framework el cual es basado en el lenguaje de Python, este *framework* genera secciones denominadas “*apps*” por cada entidad de la base de datos sobre la cual se genera integración y comunicación, estas a su vez se poseen 4 subsecciones de configuración que distribuyen y centralizan el procesamiento de la información que es enviada al backend, estas subsecciones son:

- *Models*: Estructura el modelo de la tabla o entidad de la base de datos como un objeto
- *Serializer*: Este es un esquema para decodificar las estructuras de datos provenientes de la base de datos o del mismo backend en elementos compatibles con otras tecnologías.
- *ViewSet*: Es un conjunto de herramientas que permite interactuar con la base de datos y los sistemas que requiera interconexión.
- *Urls*: Permite integrar y especificar las urls de conexión de sistemas externos.

Las *apps* creadas para la plataforma que gestionan la comunicación, almacenamiento y procesamiento de la información se presenta en la figura 29.

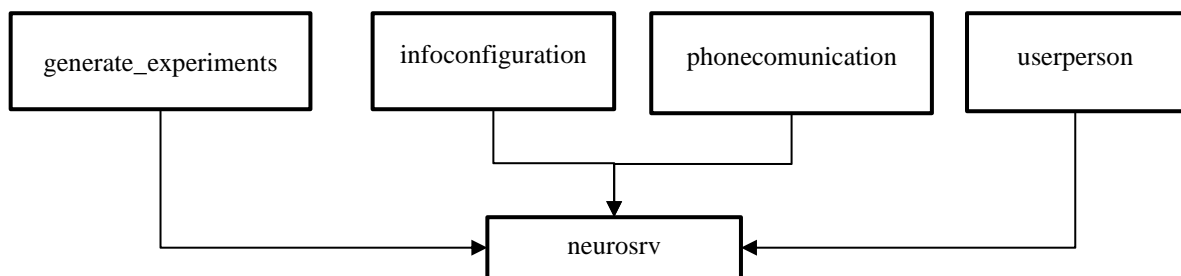


Figura 29. Diagrama de *apps* para gestión de información en el *backend*

Cada una de las *apps* presenta una función de acuerdo con las características de la plataforma la cual se describe como:

- **neurosrv**: Establece la integración de todas las *apps* que posee el backend, así como los parámetros de conexión a la base de datos y gestión de los entornos.

- **generate\_experiments:** Gestiona la inserción, actualización y obtención de los datos relacionado con los experimentos realizados tales como generación de ID, actualización de calificaciones por época y almacenamiento de la precisión del clasificador.
- **infoconfiguration:** Controla la actualización y control de las configuraciones por usuario.
- **phonecommunication:** Dirige la comunicación entre el API de Telegram y la plataforma web, así como el almacenamiento de datos de conexión.
- **userperson:** Gestiona la autenticación, creación y gestión de usuarios en la plataforma web, además de otorgar los permisos de administración de los usuarios.

Estas secciones poseen cada una de ellas *endpoints* que permiten la interacción y transmisión bidireccional API REST entre el *frontend*, interfaz de adquisición y *backend*. Para la gestión, visualización e interacción con los endpoints se implementa la librería “Swagger” que presenta una interfaz de usuario nativa y amigable para este fin, la figura 30 presenta la visualización de este entorno, el cual es posible acceder por medio de la URL del backend con el endpoint */swagger*.



Figura 30. Interfaz de usuario de *swagger* para gestión de endpoints

La plataforma de administración de usuarios la cual es proporcionada por Django Rest Framework establece un panel de control sobre el backend y permite la creación, actualización, modificación y eliminación de usuarios para el uso y funcionamiento del sistema. El acceso se realiza por medio de la URL del backend con el endpoint `/admin`, el cual al ingresar presenta la vista observada en la figura 31.

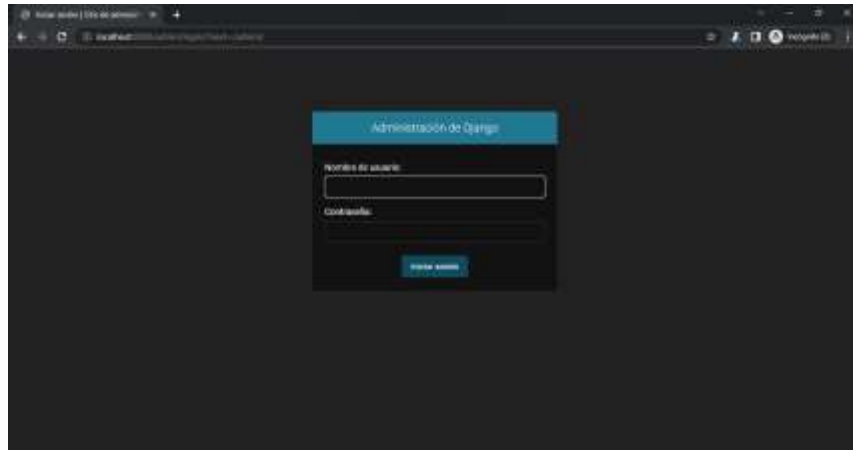


Figura 31. Interfaz de acceso a plataforma de administración de usuarios

Las credenciales de acceso para esta plataforma son las creadas como superusuario al hacer el despliegue del backend como se describe en el manual de usuario para despliegue local presentado en la sección 4.3. Al ingresar con las credenciales generadas se presenta la interfaz observada en la figura 32.

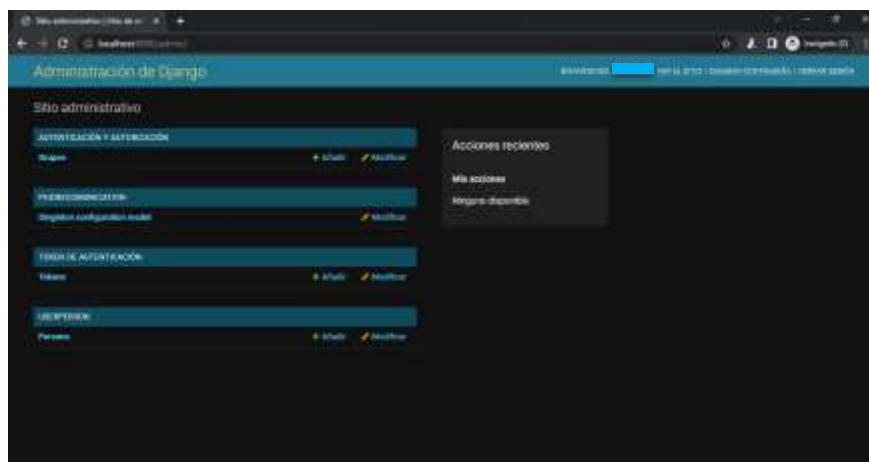


Figura 32. Vista principal de la plataforma de gestión de usuarios

Donde en la app “USERPERSON” podremos realizar la gestión de los usuarios de acuerdo con las necesidades, esta vista se puede observar en la figura 33.

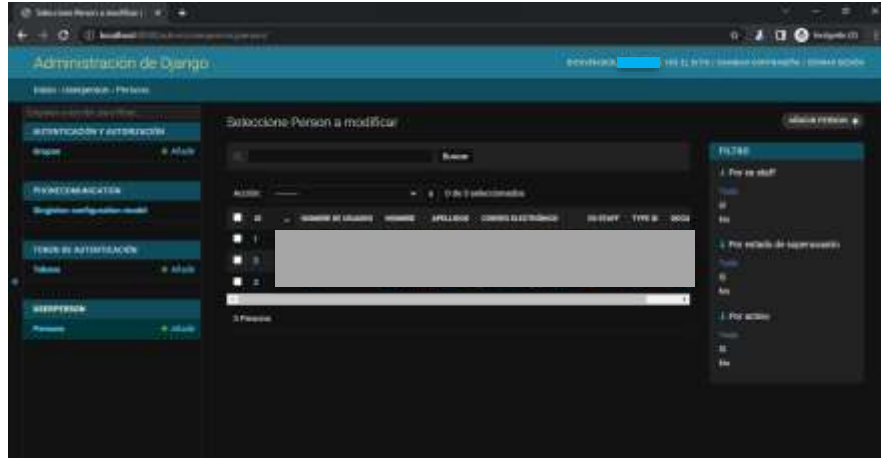


Figura 33. Vista de gestión de usuarios

El manual de usuario para el control y manejo de esta interfaz se presenta en el anexo D del presente documento.

#### 4.2.2.3 Interfaz de Adquisición

Esta interfaz a diferencia de la plataforma web no es posible almacenarla para funcionamiento en un servidor, dado que requiere del acceso a los puertos COM del dispositivo en donde se ejecute y con esto la comunicación con la tarjeta de adquisición de señales. Por medio de la comunicación con la plataforma web por medio de API REST la interfaz extrae la configuración almacenada para el usuario con respecto al ID del experimento que se ingrese, en donde ejecuta los algoritmos necesarios para la adquisición, filtrado, procesamiento, clasificación y envío de los datos relacionados con las señales se obtienen del puerto COM. La figura 34 se observa la interfaz de adquisición de señales al ejecutar el *script* asociado.



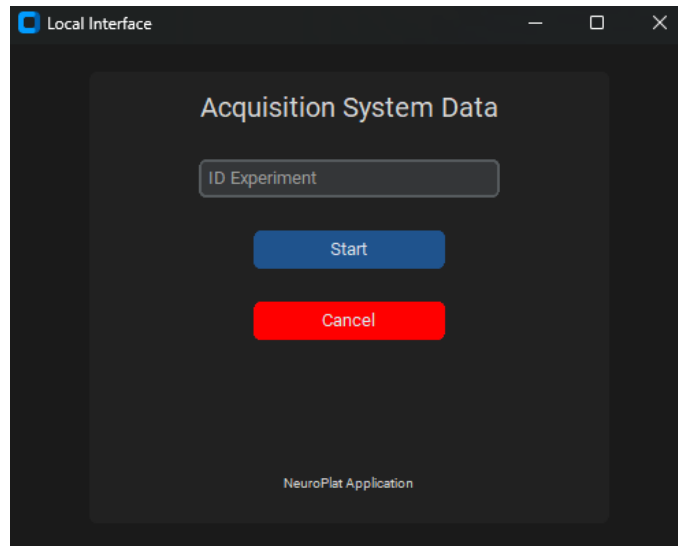


Figura 34. Vista de interfaz de adquisición de señales

Al ingresar el ID del experimento generado por la plataforma web en la sección “Experiments”, la interfaz gráfica genera la conexión con la base de datos por medio del backend y sus endpoint correspondientes, obteniendo la información de la hora de inicio del experimento y la información de las configuraciones almacenadas por el usuario. El manual de usuario de acuerdo con los mensajes de error y visualización de información para esta interfaz se presenta en el anexo E.

Posterior a la obtención de los parámetros, se genera las figuras para la visualización de las señales y la FFT de la mismas, estas figuras iniciarían la visualización de datos únicamente cuando la hora del experimento generada por el experimento sea igual que la hora del dispositivo en donde se esté ejecutando la interfaz.

Al cumplir la hora establecida se establece la comunicación por puerto serial con la tarjeta de adquisición. Dada la documentación de OpenBCI [28] se establecen dos frecuencias de muestreo 250 Hz y 125 Hz, esta configuración se realiza por medio del parámetro “Daisy” en la función que establece dicha conexión. Dado que el procesamiento de las señales se realiza en tiempo real conforme los estímulos se van generando una mayor cantidad de muestras en el menor tiempo para ser enviada al entorno de visualización y evitar fatigas en el paciente la frecuencia de muestreo seleccionada por defecto es 250 Hz, sin embargo, en

caso de que el usuario requiera de la frecuencia adicional que posee la tarjeta es posible modificarla.

El tiempo estimado de la adquisición de muestras por parte de la tarjeta se presenta como el periodo de la frecuencia seleccionada.

$$T_1 = \frac{1}{250 \text{ Hz}} = 4 \text{ ms} \quad (1)$$

En el caso de la frecuencia adicional de 125 Hz

$$T_2 = \frac{1}{125 \text{ Hz}} = 8 \text{ ms} \quad (2)$$

Cada una de las muestras obtenidas de la tarjeta de adquisición se presenta como un array con n posiciones dependiendo de la cantidad de canales utilizados, como se muestra a continuación:

[ muestra\_canal\_1, muestra\_canal\_2, muestra\_canal\_3, muestra\_canal\_4,  
muestra\_canal\_5, muestra\_canal\_6, muestra\_canal\_7, muestra\_canal\_8]

Cada muestra se descompone por canal y son almacenadas de forma continua en un array para cada canal sobre el cual se realizará el procesamiento de las muestras. La matriz de esta descomposición se observa como:

```
Channel_1 [muestra_1, muestra_2, muestra_3, ..... muestra_n]
Channel_2 [muestra_1, muestra_2, muestra_3, ..... muestra_n]
Channel_3 [muestra_1, muestra_2, muestra_3, ..... muestra_n]
      .
      .
      .
Channel_n [muestra_1, muestra_2, muestra_3, ..... muestra_n]
```

Posteriormente se realiza al almacenamiento, se crean las ventanas para el procesamiento de las muestras y relacionado con la presencia de los componentes P300. Estos componentes se presentan de 200 a 650 ms después de la generación de los estímulos [7], por lo cual el tamaño ventana debe ser del valor máximo de la generación de los componentes junto con un tiempo de seguridad que permita incluir la presencia de la onda, bajo esta definición el tamaño de la ventana en muestras se establece como:

$$T_3 = 750 \text{ ms}$$

$$N_V = \frac{T_3}{T_1} = \frac{750 \text{ ms}}{4 \text{ ms}} = 187,5 \quad (3)$$

La cantidad de muestras de la ecuación 3 se redondea hacia el entero superior ya que las posiciones del array que contiene la muestras por canal solo aceptan número enteros para su respectiva localización del array, bajo esta relación el tamaño de la ventana es de 188 muestras. Esta relación de tamaño es aplicable para todos los canales dado que la descomposición se realiza en la misma proporción para cada uno de ellos.

Con el tamaño de la ventana definido, el sistema debe calcular la cantidad de muestras que debe desplazarse para iniciar la siguiente ventana, este valor se obtiene utilizando el tiempo que configura el usuario en la plataforma web como “*Time for stimules*” ( $T_s$ ), este variable es el tiempo de espera entre parpadeos que se expresa en milisegundo. La cantidad de muestras dada para el desplazamiento está dada por la ecuación 4 y de acuerdo con los parámetros el valor se redondea al entero menor para obtener las características de la muestra previa y no generar perdidas de información.

$$N_d = \frac{T_s}{T_1} \quad (4)$$

El diagrama presentado en la figura 35 expresa gráficamente la relación que tienen las ecuaciones 3 y 4 en las ventanas deslizantes por canal para generar el procesamiento de las señales.

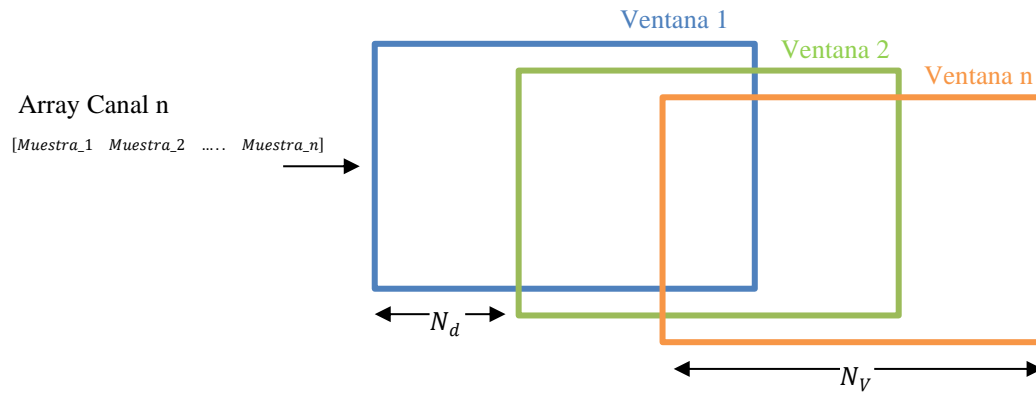


Figura 35. Diagrama de relación entre tamaño de ventana y tamaño de desplazamiento en ventanas deslizantes

Con la definición de las ventanas y el tamaño muestras de desplazamiento, para realizar el procesamiento de las muestras se deben completar como mínimo la cantidad de muestras para la primera ventana antes de realizar el procesamiento de la misma y con ella asociarla a una época. Cuando se cumplen esta cantidad de muestras el sistema procede con el filtrado, procesamiento y clasificación, cada una de las etapas es descrita a continuación.

### Filtrado

Teniendo en cuenta la figura 4 sobre el diagrama interno del módulo de filtros, se establece la construcción de los filtros digitales de acuerdo con los parámetros entregados y configurados por el usuario. Para esta sección deben construir dos filtros en cascada, el primero tipo *notch* y el segundo pasa banda desde 0.3 Hz hasta la frecuencia dada por el usuario.

Para el filtro notch requiere de dos características para su construcción: la frecuencia de filtrado y el factor de calidad, la primera se define de acuerdo con la frecuencia de la red eléctrica que se suministre a los dispositivos electrónicos, para el caso de Colombia esta frecuencia es de 60 Hz, como se especifica en la resolución 24 de 2005 [29]. La segunda

característica, el factor de calidad, es obtenido de estudios generados que presentan la relación estadística del factor de calidad del filtro a señales biomédicas, [30] bajo estas características el factor se establece en 50.[30]

El segundo filtro (filtro pasa banda) para su construcción requiere de 4 parámetros: frecuencia de corte mínima, frecuencia de corte máxima, orden del filtro y tipo de ventana. De estos parámetros la frecuencia de corte mínima es establecida en 0,2 Hz en relación a las frecuencias de las ondas delta [14], [15], la frecuencia de corte máxima y el orden del filtro son parámetros definidos por el usuario en la vista de configuración de la plataforma web y por último el tipo de ventana de acuerdo se realiza bajo la selección de componentes de la sección 4.2.1.4. Es de tener en cuenta que la frecuencia de muestreo de los filtros digitales debe ser igual a la frecuencia de adquisición de muestras por la tarjeta, por lo cual la frecuencia corte máxima dado por el teorema de Nyquist es:

$$f_{cm} = \frac{250 \text{ Hz}}{2} = 125 \text{ Hz} \quad (5)$$

La aplicación de estos filtros se realiza para cada una de las ventanas por canal conforme se van generando la adquisición de las mismas.

La función de transferencia de forma canónica para el filtro notch se describe en la ecuación 6 [31]

$$H_{fn}(s) = K \frac{s^2 + s \frac{\omega_z}{Q_z} + \omega_z^2}{s^2 + s \frac{\omega_p}{Q_p} + \omega_p^2} \quad (6)$$

Donde

$\omega_z$ : Frecuencia de cero

$\omega_p$ : Frecuencia de polo

$Q_z$ : Factor de calidad del cero

$Q_p$ : Factor de calidad del polo

La forma canónica de la función de transferencia para el filtro pasa banda se puede observar en la ecuación 7 [32]

$$H_{fb} = H_0 \frac{s^2 \left( \frac{4s}{\sqrt{\tau_0 + \frac{16}{\Delta\omega^2} - \tau_0}} \right) + \omega_0^2}{s^2 \left( \frac{4s}{\sqrt{\tau_0 + \frac{16}{\Delta\omega^2} + \tau_0}} \right) + \omega_0^2} \quad (7)$$

Las funciones de transferencia de cada uno de los filtros descrito en las ecuaciones 6 y 7 son homologables en funciones integradas en las librerías de Python.

### Procesamiento

Posterior a poseer las señales filtradas en la frecuencia correspondiente, se realiza la visualización de la Trasformada rápida de Fourier a las muestras de la ventana que permita al usuario conocer las frecuencias y su amplitud en cada canal en tiempo real. De forma paralela se realiza la aplicación de la transformada y transformada inversa de Wavelet a cada ventana por canal.

Para la aplicación de la transformada se requiere de la familia de Wavelet que se utilizara como onda pulsante [13]. La denominación de estas familias viene dada por la función utilizada pero cada una de ellas tiene un tipo de onda asociada Estas denominaciones de familias son [33]:

- Haar (haar)
- Daubechies (db)
- Symlets (sym)
- Coiflets (coif)
- Biorthogonal (bior)
- Reverse biorthogonal (rbio)
- “Discrete” FIR approximation of Meyer wavelet (dmey)

- Gaussian wavelets (gaus)
- Mexican hat wavelet (mexh)
- Morlet wavelet (morl)
- Complex Gaussian wavelets (cgau)
- Shannon wavelets (shan)
- Frequency B-Spline wavelets (fbsp)
- Complex Morlet wavelets (cmor)

Seguidas a esta denominación se describe el tipo, descrito por un numero entero si formato especifico. Un ejemplo de este formato de selección de la familia wavelet es “db6”.

La transformada de Wavelet entrega dos coeficientes necesarios para reestructurar de nuevo la señal en términos de muestras, pero con modificación de relación señal a ruido entre la presencia de los componentes P300 como elementos no periódicos y el ruido de la señal.

### **Clasificación**

Para utilizar el clasificador One Rule requiere del parámetro del porcentaje de muestras que utilizara para testeo, este valor es configurado por el usuario en la plataforma web. El cual determinara el porcentaje de muestras que utilizara para validación y el porcentaje para testeo.

Inicialmente las muestras por cada canal son separadas en 4 cuartiles (25%, 50%, 75%, 100%) el cual son el insumo del clasificador para determinar la calificación (0 a 3) de cada muestra con respecto a la dispersión y tendencia central del total de la ventana analizada. La salida de este proceso es una matriz de n posiciones (determinada de la cantidad de muestras de la ventana) por n canales establecidos, la representación de esta matriz se puede observar cómo:

<i>Canal 1</i>	[ <i>Calificación_1</i>	<i>Calificación_2</i>	...	<i>Calificación_n</i> ]
<i>Canal 2</i>	[ <i>Calificación_1</i>	<i>Calificación_2</i>	...	<i>Calificación_n</i> ]
	...	...	...	...
<i>Canal n</i>	[ <i>Calificación_1</i>	<i>Calificación_2</i>	...	<i>Calificación_n</i> ]

Esto permite identificar en que canal se presenta una calificación más alta y cuál de ellos presenta una mayor detección de componentes no periódicos en la señal.

Finalmente, el clasificador para determinar la calificación total con respecto a los  $n$  canales de la presencia de componentes P300 (no periódicos en la señal), utiliza un algoritmo de entrenamiento (elemento de *machine learning*) que procesa las valoraciones de la matriz previa en porcentajes de validación y entrenamiento, para determinar la cantidad de clases cada una de ellas con una calificación dependiendo de la relación que contenga cada muestra de cada canal con la clase especificada. El promedio de las calificaciones por clase, será utilizado para determinar si la época presenta o no componentes P300. Cada ventana presenta una valoración que se va guardando para ser enviada a la matriz de estímulos. El array que contiene las calificaciones por época dadas por el clasificador se expresa como:

Carácter 1	[ <i>Calf. Epoca 1</i> <i>Calf. Epoca 2</i> <i>Calf. Epoca 3</i> <i>Calf. Epoca 4</i> ....]
Carácter 2	[ <i>Calf. Epoca 1</i> <i>Calf. Epoca 2</i> <i>Calf. Epoca 3</i> <i>Calf. Epoca 4</i> ....]
...	
Carácter $n$	[ <i>Calf. Epoca 1</i> <i>Calf. Epoca 2</i> <i>Calf. Epoca 3</i> <i>Calf. Epoca 4</i> ....]

### Identificación

Por medio de la conexión de un *Web Socket* entre el *FrontEnd* y el *Backend*, cuando la interfaz de adquisición realice el procesamiento y clasificación enviara el array de calificaciones por época a esta conexión para almacenarla y enviarla a la vista *matrix*.

La vista *matrix* genera de forma aleatoria el número de la columna o fila a iluminar, el cual intercambia entre fila o columna iniciando siempre con columna. El numero generado de forma aleatoria (época generada) es almacenado en una variable de tipo array (con  $n$  posiciones dependiendo de la cantidad de estímulos que estipule el usuario) para generar la identificación con respecto a la calificación por época entregada por la interfaz de adquisición. La especificación de la variable que almacena las posiciones con respecto a la cantidad de letras dadas por el usuario puede visualizar como



	Column	Fila	Column	Fila	
	↓	↓	↓	↓	
Carácter 1	[ <i>Epoca 1</i>	<i>Epoca 2</i>	<i>Epoca 3</i>	<i>Epoca 4</i>	....]
Carácter 2	[ <i>Epoca 1</i>	<i>Epoca 2</i>	<i>Epoca 3</i>	<i>Epoca 4</i>	....]
	...				
Carácter n	[ <i>Epoca 1</i>	<i>Epoca 2</i>	<i>Epoca 3</i>	<i>Epoca 4</i>	....]

Al obtener los datos obtenidos de la interfaz de adquisición se procede a asignar cada calificación de épocas con respecto a la posición de época en la matriz separándolas entre filas y columnas, finalizando con identificar la posición de las mismas que poseen mayor calificación de época, y visualizando la letra identificada en la matriz por 2,5 segundos. Al finalizar la vista de matriz envía la comunicación a la interfaz para iniciar nuevamente con la siguiente carácter o secuencia. Al hacer la identificación por parte de la matriz se visualiza en la vista de Statistics y transmitida a la aplicación del dispositivo móvil, para completar la transmisión completa.

#### 4.2.3 Validación

La validación del funcionamiento de la plataforma, interfaz, comunicaciones y almacenamiento de todos y cada uno de las secciones del sistema se presenta un plan de pruebas expuesto en el anexo H.

### 4.3 Descripción Técnica del Producto

#### Modelo UML

Los modelos UML asociados a las principales vistas de la plataforma web que requieren sistema de control por roles se observan en las figuras 36 a la 39.

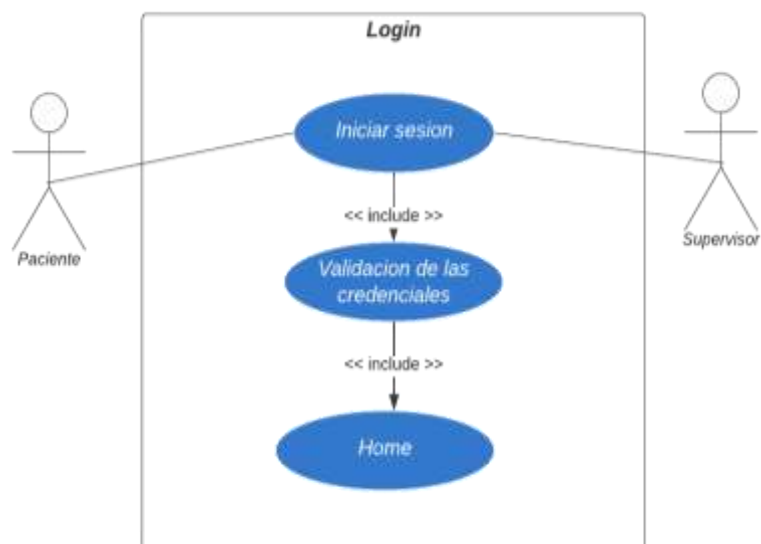


Figura 36. Modelo UML para la vista Login e interacción entre el paciente y supervisor

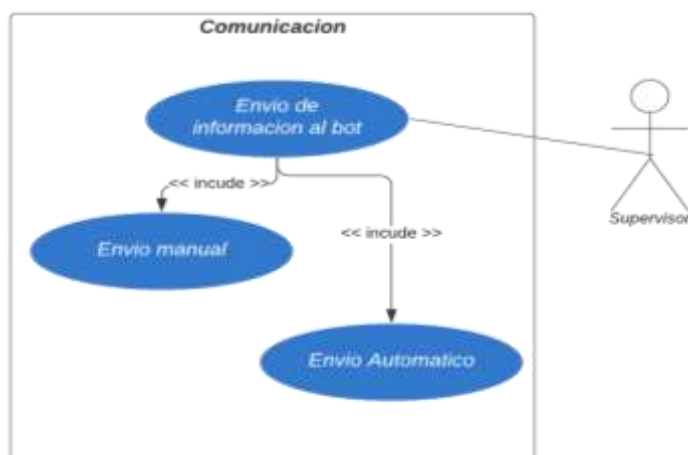


Figura 37. Modelo UML para la vista "Communication" entre la plataforma web y el dispositivo móvil

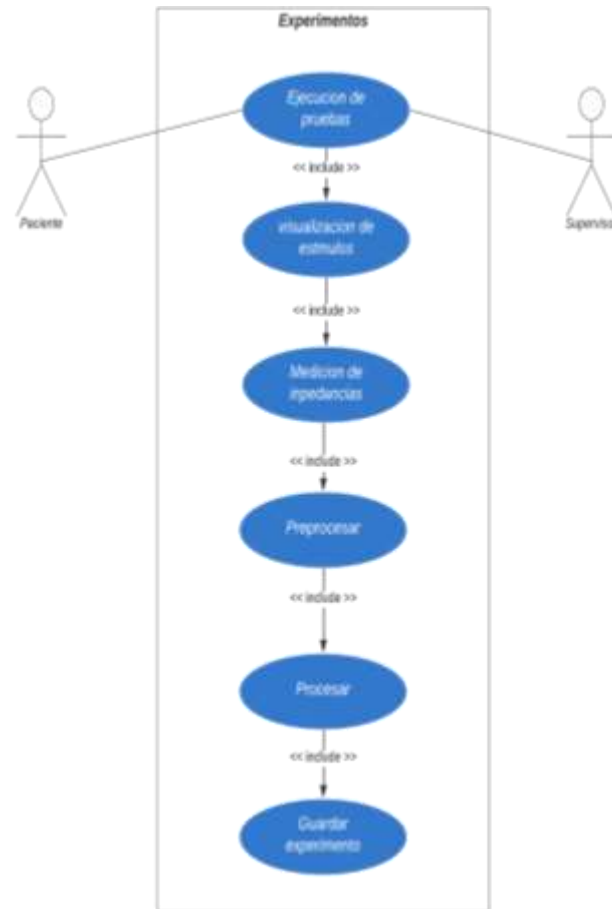


Figura 38. Modelo UML para la vista “Experimentos” con respecto al generador de estímulos y el supervisor de los procedimientos

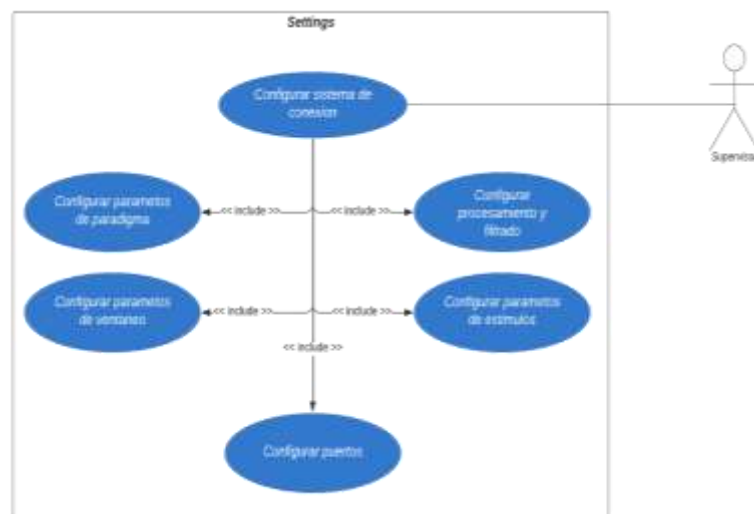


Figura 39. Modelo UML para la vista “Settings” bajo el rol de supervisor

## Diagrama de flujo de funcionamiento del sistema

El diagrama de flujo respecto al funcionamiento del sistema con respecto al almacenamiento de información, generación de experimentos y comunicación entre sistemas se presenta en la imagen 40.

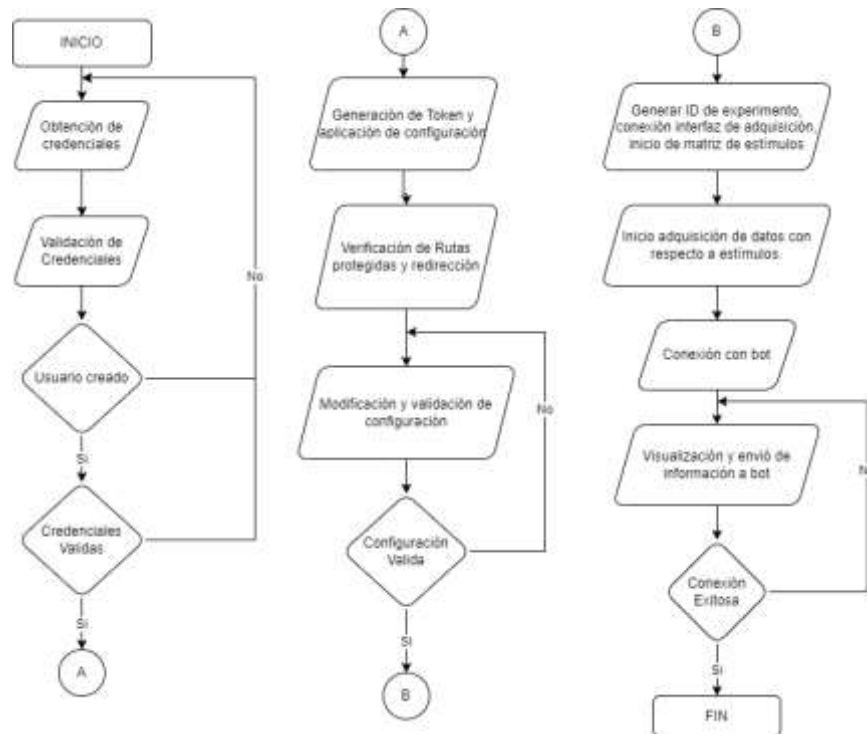


Figura 40. Diagrama de Flujo general del sistema

## Arquitectura de la aplicación

La interfaz se basa en una arquitectura dividida en tres secciones las cuales se especifican como:

- *FrontEnd Application*: Contiene todos la visualización e interacción del usuario con el sistema, se basa en el *framework* React Js y en el *meta-framework* Next Js.
- *NeuroPlat Services*: Contiene la interacción entre el *frontEnd* y *Backend* que incluye el procesamiento, filtrado y configuración de las señales a tratar, el *meta-framework* utilizado para esta seccion es Django REST Framework

- *BackEnd Application*: Contiene todas las características para guardar y obtener la información de la base de datos que se le suministra al usuario para su debida visualización. El *framework* utilizado es Django con la integración de Postgress como base de datos

Todas las anteriores secciones estas incluidas dentro del servicio AWS en el cual se desplegará la aplicación con todos sus componentes, requerimientos y restricciones. La visualización de la presente arquitectura se presenta en el anexo A del presente documento.

### **Modelo Entidad-Relación**

El modelo entidad relación de la base datos del sistema se presenta en el anexo B del presente documento.

### **Manuales de Usuario**

- **Manual de Usuario para despliegue local de aplicaciones (FrontEnd y BackEnd)**

Para el desarrollo, actualización, funcionamiento y verificación de información de la plataforma web y la interfaz de adquisición de forma local (Funcionamiento en computador local) se debe seguir el procedimiento correspondiente para cada una de los entornos del sistema, lo anterior se describe en un archivo README.md que se almacena con el código en la plataforma de control de versiones. Las urls de acceso de estos archivos se describen a continuación:

#### **Despliegue Local *FrontEnd***

**URL:** <https://github.com/jcoronadop/NeuroPlat/blob/master/app/README.md>

#### **Despliegue Local *BackEnd***

**URL:** <https://github.com/jcoronadop/NeuroPlat/blob/master/srv/README.md>

➤ **Manuales de Usuario control de las plataformas, interacción del usuario con las interfaces y tarjeta de adquisición de señales.**

El manual de usuario para el correcto funcionamiento, implementación y adquisición de señales de la tarjeta por integración se especifica en el anexo C.

El manual de usuario para de la interacción del usuario con la plataforma web y la plataforma de administración de usuarios se indica en el anexo D.

El manual de usuario para el manejo de la interfaz que genera la adquisición de señales por la tarjeta se especifica en el anexo E.

## 5. RESULTADOS Y ANÁLISIS DE RESULTADOS

Los procedimientos realizados al sistema se llevaron a cabo con tres voluntarios bajo un protocolo específico el cual se describe a continuación.

### 5.1 Protocolo de ejecución de procedimientos

Las pruebas de deletreo para fase de entrenamiento de los voluntarios fueron realizadas bajo el siguiente procedimiento:

- Determinación de espacio para generación de pruebas
- Determinación de condiciones de iluminación del espacio
- Explicación a cada voluntario sobre el proceso de ejecución y duración estimada de cada prueba.
- Preparación de elementos y espacio para ejecución de pruebas
- Conexión de electrodos al voluntario y tarjeta de adquisición ordenador.
- Establecimiento de parámetros y configuraciones para pruebas.
- Ejecución de pruebas de deletreo por entrenamiento
- Almacenamiento y visualización de datos
- Desconexión de elementos al paciente
- Limpieza de elementos
- Finalización de pruebas

### 5.2 Características de los voluntarios

Para la ejecución del protocolo de pruebas descrito en el numeral 5.1. en la presente sección se especifican las características de cada voluntario, descritas en la tabla 10.

Tabla 10. Características por voluntario

Voluntario	Edad	Genero	Condición o Enfermedad Asociada
A	27	Masculino	Ninguna
B	25	Femenino	Ninguna
C	26	Masculino	Ninguna

### 5.3 Características de las pruebas

Para la ejecución de las pruebas correspondientes por voluntario se establecen las características por prueba, las cuales se describen en la tabla 11.

Tabla 11. características de las pruebas ejecutadas

Voluntario	Iluminación	Tiempo entre parpadeos (ms)	Cantidad de parpadeos	Porcentaje de muestras para entrenamiento de clasificador
A	Clara	100	40	20%
B	Clara	100	40	33%
C	Oscura	100	40	40%

Además, para estas características todos los voluntarios deletrearon un total de 5 caracteres por 5 secuencias. Las configuraciones por prueba utilizadas para las pruebas se describen en la figura , con la configuración por paciente del parámetro “*Size data test*” de acuerdo con lo descrito en la tabla 11.

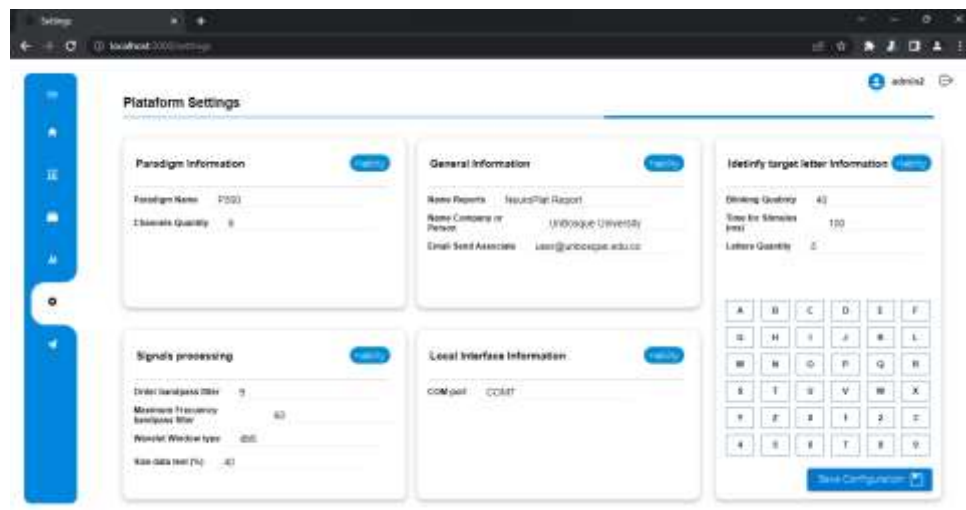


Figura 41. Configuración base para ejecución de pruebas

### 5.4 Preparación y conexión de elementos a voluntarios

Para cada uno de los pacientes se les fue informado el proceso de conexión de los electrodos, procedimiento de ejecución de las pruebas, el cual consiste en el pensamiento de la letra indicada por el sistema (Determinada de forma aleatoria con los parámetros de matriz



configurados) durante el espacio destinado para la misma. La figura 42, permite visualizar la conexión de los electrodos por cada voluntario.

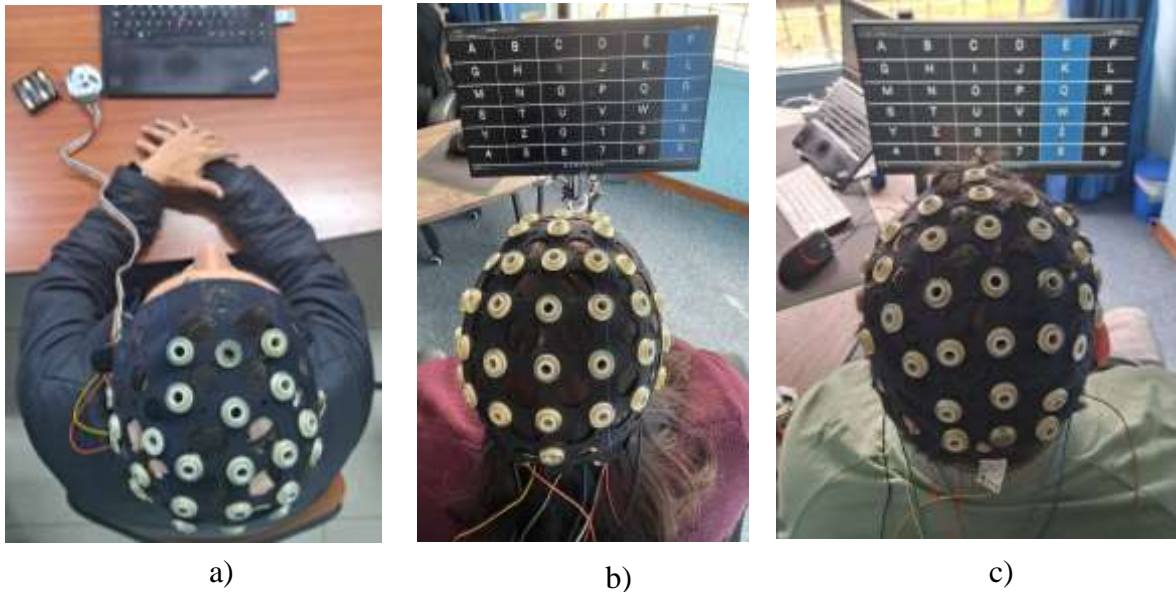


Figura 42. Conexión de electrodos a voluntarios para ejecución de pruebas a) Voluntario A b) Voluntario B c) Voluntario C

### 5.5 Visualización de señales

Posterior a la conexión de los electrodos y la tarjeta de adquisición al ordenador se procede en cada secuencia a generar el ID del experimento de acuerdo con lo expresado en el anexo D para la ejecución de estos. La figura 43 permite observar las señales de los pacientes en un instante de muestras, además las figura 44 expresa la FFT para las señales por canal obtenidas de la tarjeta de adquisición.

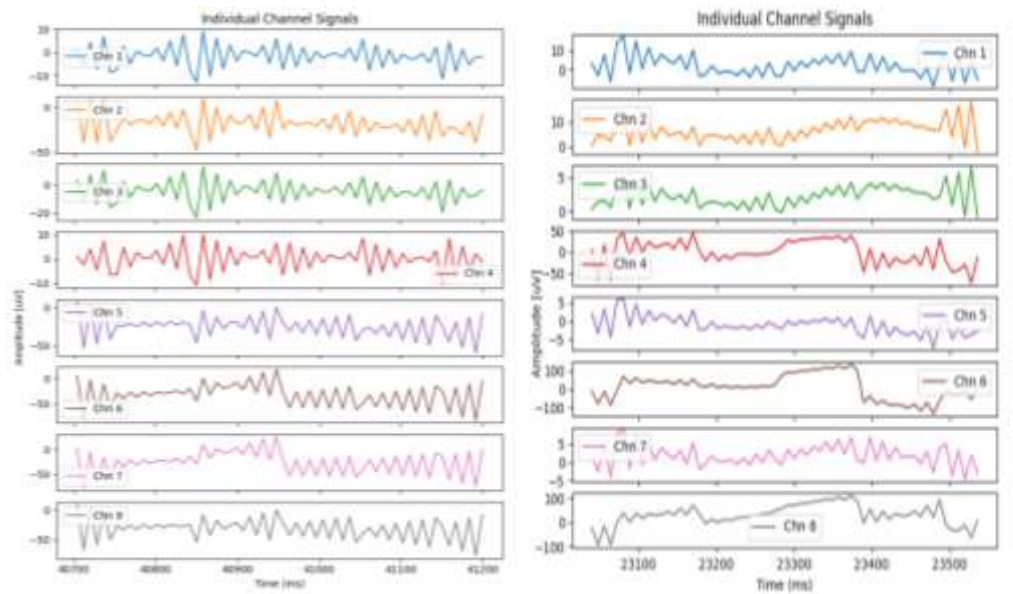


Figura 43. Visualización de señales por canal en un conjunto de muestras, en cada uno de los electrodos en diferentes instantes

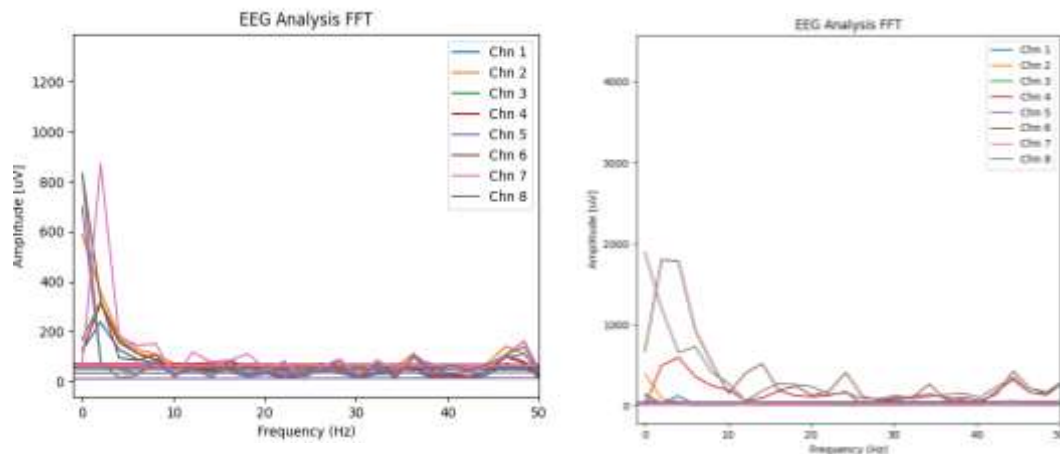


Figura 44. FFT del conjunto de canales de las señales en el dominio del tiempo en diferentes instantes

## 5.1 Relación de datos obtenidos

Posterior a la ejecución de cada prueba en cada voluntario se procede con la relación de información resultante de las pruebas ejecutadas. Las tablas 12, 13 y 14 expresan la información por cada uno de los voluntarios

Tabla 12. Resultados de experimentos para el voluntario A

Secuencia	Promedio calificación por carácter	Promedio precisión del clasificador por carácter	Carácter generado por el sistema	Carácter deletreado
<b>1</b>	0,285	39,5%	X	R
	0,329	40,0%	T	Z
	0,389	45,9%	I	2
	0,424	48,0%	6	0
	0,404	41,0%	4	Q
<b>2</b>	0,349	38,0%	9	B
	0,404	40,5%	X	I
	0,435	42,0%	H	I
	0,344	40,5%	Z	U
	0,430	40,5%	F	6
<b>3</b>	0,445	41,0%	L	C
	0,444	45,0%	2	3
	0,365	41,4%	W	R
	0,464	44,5%	H	Q
	0,670	51,5%	9	H
<b>4</b>	0,635	48,5%	E	L
	0,690	52,5%	2	2
	0,580	48,9%	X	X
	0,579	48,5%	8	X
	0,705	53,5%	K	D
<b>5</b>	0,820	52,0%	R	R
	0,905	58,5%	9	Q
	0,940	55,5%	S	H
	0,815	57,0%	E	W
	0,700	56,4%	M	W

Las concordancias entre el carácter generado por el sistema y deletreado por el voluntario A se presentaron en la secuencia 4 y 5, las cuales identifica con el color verde en la tabla 12, con un promedio global de calificación por carácter de 0,542 y promedio global de la precisión del clasificador de 46,8 %.

Tabla 13. Resultados de experimentos para el voluntario B

Secuencia	Promedio calificación por carácter	Promedio precisión del clasificador por carácter	Carácter generado por el sistema	Carácter deletreado
<b>1</b>	0,796	42,9%	9	I
	0,970	47,0%	S	P
	0,891	43,3%	I	O
	0,950	42,9%	U	5
	1,016	46,3%	F	9
<b>2</b>	0,941	45,4%	P	B
	0,979	46,6%	B	J
	0,979	47,9%	M	5
	0,829	45,0%	R	K
	1,000	48,3%	B	O
<b>3</b>	0,941	47,0%	9	B
	0,966	48,3%	F	V
	0,975	47,5%	7	5
	1,016	47,9%	W	L
	0,933	46,2%	R	B
<b>4</b>	1,000	49,0%	0	X
	0,995	49,1%	O	D
	0,966	48,7%	Q	I
	1,112	50,8%	Z	A
	1,125	50,8%	<b>F</b>	<b>F</b>
<b>5</b>	1,029	48,7%	F	O
	0,974	46,6%	<b>P</b>	<b>P</b>
	0,904	47,0%	C	I
	1,000	49,5%	Z	D
	0,950	47,9%	T	U

Las concordancias entre el carácter generado por el sistema y deletreado por el voluntario B se presentaron en la secuencia 4 y 5, las cuales identifica con el color verde en la tabla 13, con un promedio global de calificación por carácter de 0,969 y promedio global de la precisión del clasificador de 47,2 %.

Tabla 14. Resultados de experimentos para el voluntario C

Secuencia	Promedio calificación por carácter	Promedio precisión del clasificador por carácter	Carácter generado por el sistema	Carácter deletreado
1	0,756	65,6%	I	N
	0,756	68,7%	U	N
	0,737	70,5%	5	T
	0,775	66,8%	6	C
	0,731	69,3%	E	R
2	0,762	71,8%	X	T
	0,756	66,2%	3	F
	0,713	68,7%	J	0
	0,768	73,1%	X	X
	0,775	71,8%	U	E
3	0,781	70,6%	Y	H
	0,800	71,8%	L	S
	0,737	71,2%	8	G
	0,737	73,1%	2	2
	0,678	70,0%	K	G
4	0,790	69,3%	3	I
	0,785	71,8%	W	W
	0,818	69,3%	3	I
	0,675	67,5%	V	S
	0,768	72,5%	A	D
5	0,656	63,7%	8	J
	0,675	72,5%	Z	Z
	0,731	71,3%	H	H
	0,650	65,0%	3	F
	0,756	63,8%	3	7

Las concordancias entre el carácter generado por el sistema y deletreado por el voluntario C se presentaron en todas las secuencias, las cuales identifica con el color verde en la tabla 13, con un promedio global de calificación por carácter de 0,743 y promedio global de la precisión del clasificador de 69,7 %.

La precisión del clasificador y la calificación por carácter en relación a cada voluntario se puede visualizar en la figura 45 y 46 respectivamente.

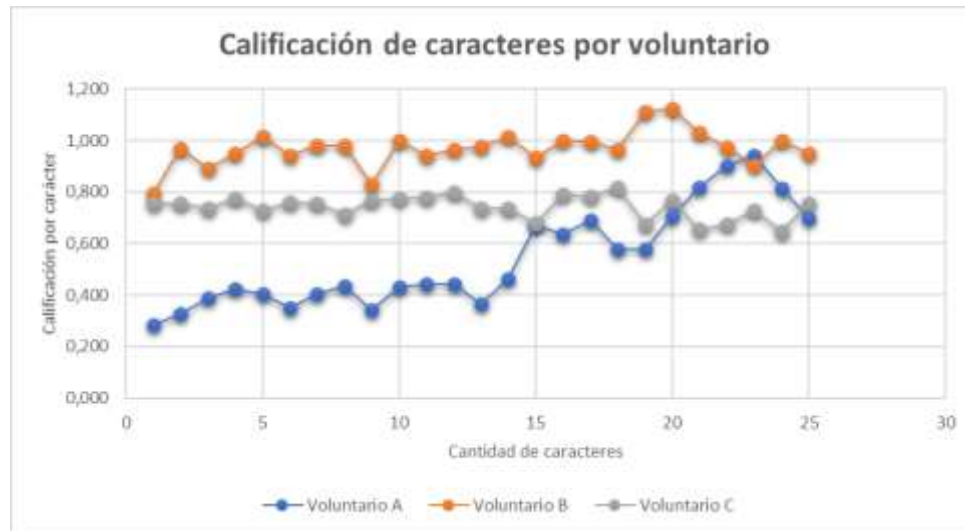


Figura 45. Grafica de calificación de época por voluntario

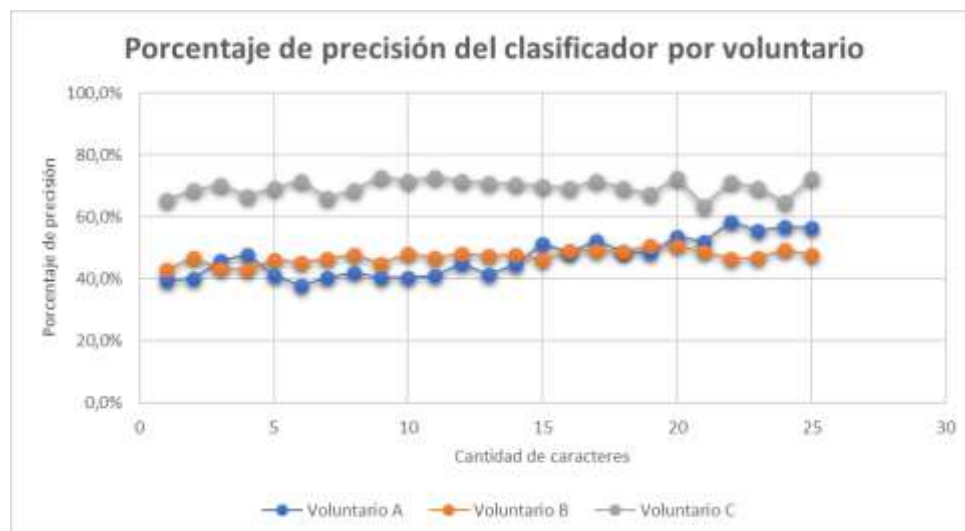


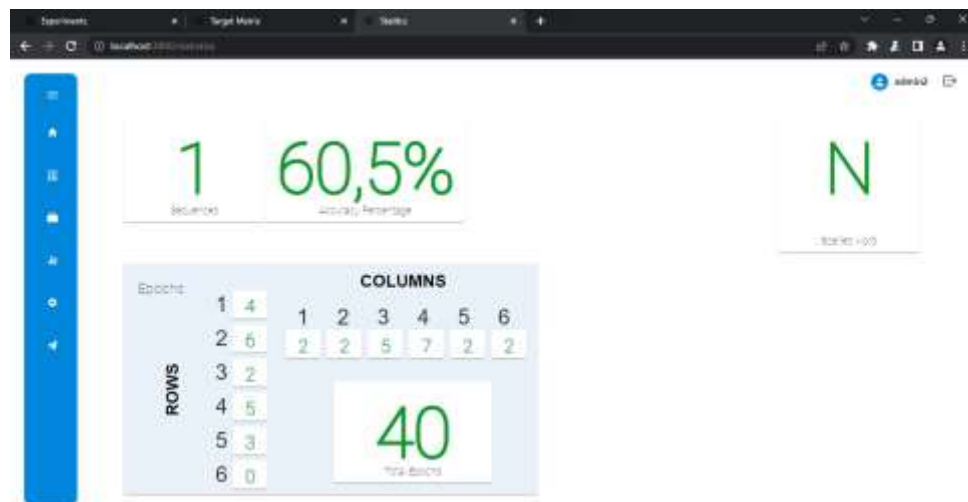
Figura 46. Grafica de precisión del clasificador por voluntario

Con los datos previamente, la tabla 15 presenta los resultados por voluntario con respecto a la cantidad de caracteres correctamente deletreado y la precisión del sistema en relación a las características de la prueba y el voluntario.

Tabla 15. Relación de precisión del sistema para cada voluntario

Voluntario	Cantidad de caracteres totales	Caracteres correctos deletreados	Precisión sistema
A	15	3	20,0 %
B	15	2	13,3 %
C	15	5	33,3 %

Posterior a realizar la visualización de los caracteres en la matriz de acuerdo con las calificaciones obtenidas por carácter, se procede con la visualización de las épocas con respecto a las secuencias, la precisión del clasificador y las letras identificadas en la vista de *Statistics*, como se puede observar en la figura 47.

Figura 47. Visualización de información respecto al desarrollo de un procedimiento en la vista *Statistics*

Finalmente, la información es transmitida al *bot* por medio de la integración realizada en la vista *Communication*, la cual previamente se realiza la configuración y verificación en comunicación como se indica en el anexo D del presente documento. La figura 48 permite visualizar la información generada en la secuencia obtenida de la figura 47.

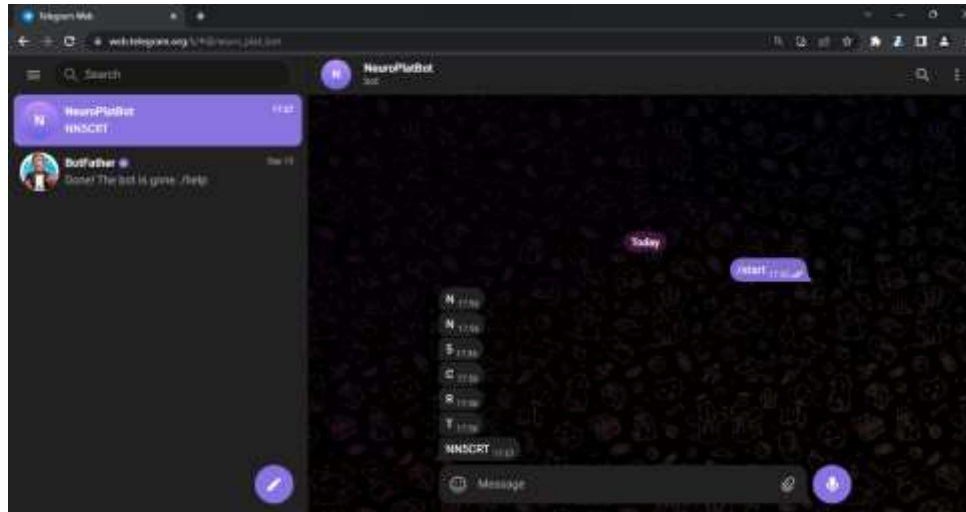


Figura 48. Comunicación de información desde la plataforma web al bot de Telegram con la información correspondiente

Según los resultados presentados previamente es posible analizar:

Las características de las pruebas y de los participantes se establecieron de forma homogénea con el fin de disminuir variables no controladas que puedan afectar los resultados de verificación, algoritmos implementados y secuencia lógica del sistema. De igual forma dadas las condiciones de las pruebas, la cantidad de secuencias realizadas por cada voluntario presento aumento en la cantidad de letras deletreadas correctamente por secuencia conforme se van ejecutando un número mayor de secuencias, independiente del porcentaje de muestras para el algoritmo de entrenamiento del clasificador.

Respecto a las calificaciones entregadas por el clasificador en cada época o carácter deletreado no varían de forma abrupta en comparación entre épocas de la misma secuencia, con una desviación máxima entre épocas de la misma secuencia de 11,4 %, estableciendo una base de precisión sobre la implementación del elemento de *machine learning* (algoritmo de entrenamiento del clasificador OneRule) y el correcto funcionamiento de este con las técnicas de procesamiento y ventaneo de las muestras implementadas.

La precisión del clasificador aumenta conforme el porcentaje de muestras configurado para el algoritmo de entrenamiento, pasando del 46,8% de precisión con un porcentaje de muestras del 20% al 69,7% de precisión con el porcentaje de muestras del 40%, como se puede apreciar



en la figura 46. Adicionalmente la precisión del clasificador no presenta alteraciones entre secuencias o entre delecteo de caracteres con una desviación estándar máxima de 6,36 %.

Es fundamental destacar la visualización de las señales por canal en el dominio del tiempo y la transformada rápida de Fourier (FFT), expresadas en las figuras 43 y 44 que permite a los usuarios observar de manera inmediata el comportamiento de las mediciones obtenidas a través de la tarjeta de adquisición de datos. Lo anterior resulta indispensable para verificar el correcto funcionamiento del sistema y la visualización de las señales con respecto a las frecuencias que contiene, que permite la identificación de patrones relevantes antes, durante y después de la ejecución de los experimentos.

De acuerdo con los datos entregados en la tabla 15, la precisión del sistema aumenta conforme el voluntario aumenta la cantidad de pruebas realizadas y bajo una configuración del porcentaje para el algoritmo de entrenamiento mayor al 33%, sin embargo, no se han contemplado variables adicionales tales como el orden y frecuencia máxima de corte del filtro pasa banda, tipo de familia de la transformada de wavelet, cantidad de canales, configuración de tiempos de parpadeo y cantidad de estímulos generados por carácter, lo que puede generar patrones adicionales que modifique la precisión del sistema de acuerdo con los voluntarios.

## 6. CONCLUSIONES

El sistema de adquisición realiza la adquisición de señales al usuario por medio de métodos no invasivos, pero asegurando la correcta y adecuada instalación de los mismos de acuerdo con las indicaciones del fabricante de la tarjeta.

El sistema garantiza el procesamiento de componentes P300 de acuerdo con la duración prevista de estas ondas, realizado en los procesos de ventaneo previos al filtrado, procesamiento, clasificación y entrenamiento de las muestras utilizadas.

Se incorporaron algoritmos para la adquisición, procesamiento, caracterización y clasificación de los componentes P300. Estos algoritmos proporcionaron los datos necesarios para el correcto funcionamiento del sistema. Dentro del algoritmo de clasificación se integra un componente de *machine learning* que contribuye a aumentar la precisión del clasificador conforme se realiza el ventaneo de las muestras, conforme a los resultados obtenidos.

La comunicación entre el *bot* de Telegram y la plataforma web se realiza de forma asíncrona, la cual no requiere de periodos específicos de funcionamiento del sistema para la transmisión y recepción, esta comunicación se realiza por medio del protocolo TCP/IP integrado en la API de Telegram, proceso ejecutado en paralelo al proceso generación, visualización y configuración de los estímulos o de la plataforma, garantizando la correcta comunicación entre los sistemas y configuración dinámica para el usuario de los *bots* utilizados.

La plataforma web integro procesos de seguridad tales como: token de acceso para la autenticación de usuarios y administración por medio de la plataforma correspondiente con acceso únicamente por superusuario. Además, permite a los usuarios realizar configuraciones sobre los procedimientos a ejecutar mejorando así la adaptabilidad de los procedimientos a pacientes o voluntarios específicos.

## 7. RECOMENDACIONES

Teniendo en cuenta el alcance, los requerimientos, limitaciones y resultados obtenidos del presente proyecto a continuación se presentan varias recomendaciones con respecto al desarrollo o integración de interfaces BCI.

Es posible Incluir múltiples paradigmas relacionados a la detección específica de características de la señal, como por ejemplo P200 o potenciales de estado estacionario, donde el usuario pueda elegirlo y usarlo en sus procedimientos, debido al dinamismo de la configuración y adaptabilidad de los lenguajes de programación utilizados.

Realizar el escalamiento de los sistemas de *FrontEnd* y *Backend* para soportar canales multiusuario, almacenamiento base de muestras tomadas o integración con otras tecnologías que se presenten en el mercado

Incluir estímulos auditivos, táctiles o sensoriales específicos dentro de la matriz de estímulos con respecto a las necesidades del paradigma, técnica, algoritmo o usuario que lo requiera.

Integrar más de un tipo, fabricante, modelo o referencia de tarjeta de adquisición de señales electroencefalográficas a la plataforma web, con respecto a las necesidades del usuario

Incorporar más de un método, técnica o algoritmo de detección de letras *tarjet* que mejore la precisión y exactitud del sistema. Además de procesos de machine learning adicionales que permita mejorar la precisión del sistema.

## 8. REFERENCIAS BIBLIOGRÁFICAS

- [1] J. Shih, D. Krusienski, y J. Wolpaw, “Brain-Computer Interfaces in Medicine”, *Mayo Clinic proceedings. Mayo Clinic*, vol. 87, pp. 268–279, ago. 2012, doi: 10.1016/j.mayocp.2011.12.008.
- [2] T. Fang *et al.*, “Recent Advances of P300 Speller Paradigms and Algorithms”, en *2021 9th International Winter Conference on Brain-Computer Interface (BCI)*, 2021, pp. 1–6. doi: 10.1109/BCI51272.2021.9385369.
- [3] V. A. Bulanov, A. V Zakharov, y E. V Khivintseva, “Wavelet transform for the identification of P300”, *IOP Conf Ser Mater Sci Eng*, vol. 862, núm. 5, p. 52049, may 2020, doi: 10.1088/1757-899X/862/5/052049.
- [4] J. Ramirez-Cortes, V. Alarcon-Aquino, G. Rosas-Cholula, P. Gomez-Gil, y P. J. Escamilla-Ambrosio, “P-300 Rhythm Detection Using ANFIS Algorithm and Wavelet Feature Extraction in EEG Signals”, *Lecture Notes in Engineering and Computer Science*, vol. 2186, ago. 2010.
- [5] “Prototipo de equipo médico de Neurofeedback para videojuego controlado con potenciales evocados”. [En línea]. Disponible en: <http://hdl.handle.net/20.500.12495/8026>
- [6] Z. Oralhan, “A New Paradigm for Region-Based P300 Speller in Brain Computer Interface”, *IEEE Access*, vol. PP, p. 1, ago. 2019, doi: 10.1109/ACCESS.2019.2933049.
- [7] C. E.-J. J. V. M. L. V. Q. M. V. L. Barbosa, “COGNITIVE FUNCTION USING EVENT-RELATED POTENTIALS (P300) ON EPILEPTIC PATIENTS: A COLOMBIAN STUDY”, Universidad de los Andes, Bogotá D.C., 2019.
- [8] Eliana García Cossio, Carlos Fernandez, Maria Eugenia Gaviria, Catalina Palacio, Liliana Alvarán, y Róbinson Alberto Torres Villa, “Interfaz cerebro computador basada en P300 para la comunicación alternativa: estudio de caso en dos adolescentes en situación de discapacidad motora”, Enviado - Antioquia, 2011.
- [9] M. G. Navarrete, “Estudio y análisis de la teoria wavelet enfocado al tratamiento de señales unidimensionales”, Universidad Pontificia Bolivariana, Bucaramanga, 2008.

- [10] Universidad Nacional del centro de la provincia de Buenos Aires, “Introducción a la Transformada Wavelet - DESCOMPOSICIÓN DE SEÑALES”, 2006.
- [11] R. A. Gonzales, “Algoritmo basado en Wavelets aplicado a la detección de incendios forestales”, Universidad de las Américas de Puebla, Puebla, Mexico, 2010.
- [12] F. Santamaría, C. A. Cortés, y F. J. Román, “Uso de la Transformada de Ondeletas (Wavelet Transform) en la Reducción de Ruidos en las Señales de Campo Eléctrico producidas por Rayos”, *Información tecnológica*, vol. 23, núm. 1, pp. 65–78, 2012, doi: 10.4067/S0718-07642012000100008.
- [13] M. A. Alzate, “Uso de la Transformada Wavelet para el estudio de tráfico fractal en redes de comunicaciones”, Bogotá D.C., Colombia, 2002.
- [14] IMOTIONS, “Neural Oscillations – Interpreting EEG Frequency Bands”.
- [15] M. Abo-Zahhad, S. Ahmed, y S. N. Seha, “A New EEG Acquisition Protocol for Biometric Identification Using Eye Blinking Signals”, *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 07, pp. 48–54, sep. 2015, doi: 10.5815/ijisa.2015.06.05.
- [16] M. S. Sappia, “Estudio comparativo de algoritmos de clasificación de señales electroencefalográficas en el paradigma del deletreador P300”, Córdoba, 2018.
- [17] J. L. Sirvent, J. M. Azorín, E. Iáñez, A. Úbeda, y E. Fernández, “Interfaz Cerebral no Invasiva basada en Potenciales Evocados para el Control de un Brazo Robot”, *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 8, núm. 2, pp. 103–111, abr. 2011, doi: 10.1016/S1697-7912(11)70031-0.
- [18] J. Polich, “Updating P300: an integrative theory of P3a and P3b.”, *Clin Neurophysiol*, vol. 118, núm. 10, pp. 2128–48, oct. 2007, doi: 10.1016/j.clinph.2007.04.019.
- [19] J. Polich y A. Kok, “Cognitive and biological determinants of P300: an integrative review”, *Biol Psychol*, vol. 41, núm. 2, pp. 103–146, oct. 1995, doi: 10.1016/0301-0511(95)05130-9.
- [20] S. Venate y T. D. Sunny, “Multiclass Classification of EEG Signal Using a Probabilistic Approach”, *Procedia Technology*, vol. 24, pp. 1002–1007, 2016, doi: 10.1016/j.protcy.2016.05.219.

- [21] S. Kundu y S. Ari, “Brain-Computer Interface Speller System for Alternative Communication: A Review”, *IRBM*, vol. 43, núm. 4, pp. 317–324, ago. 2022, doi: 10.1016/j.irbm.2021.07.001.
- [22] Hornero. R. Corralejo. D. Á. Roberto, “Brain-Computer Interface (bCI) aplicado al entrenamiento cognitivo y control domótico para prevenir los efectos del envejecimiento”.
- [23] GeeksforGeeks, “Learn-One-Rule Algorithm”, <https://www.geeksforgeeks.org/learn-one-rule-algorithm/>.
- [24] Sebastian Raschka, “OneRClassifier: One Rule (OneR) method for classification”.
- [25] Dr. Saed Sayad, “OneR”, <https://www.saedsayad.com/oner.htm>.
- [26] B. Roy *et al.*, *Ley 1581 de 2012*. Colombia: Diario oficial del departamento de función pública, 2012.
- [27] Ministerio de Salud de Colombia, *Normas científicas, técnicas y administrativas para la investigación en salud*. . Colombia: Ministerio de Salud, 1993.
- [28] OpenBCI, “OpenBCI Documentation”.
- [29] Comisión de Regulación de energía y gas, *RESOLUCIÓN 24 DE 2005*. Colombia: [https://gestornormativo.creg.gov.co/gestor/entorno/docs/resolucion\\_creg\\_0024\\_2005.htm#1](https://gestornormativo.creg.gov.co/gestor/entorno/docs/resolucion_creg_0024_2005.htm#1), 2005.
- [30] R. Sameni, “A linear Kalman Notch Filter for Power-Line Interference Cancellation”, en *AISP 2012 - 16th CSI International Symposium on Artificial Intelligence and Signal Processing*, sep. 2012, pp. 604–610. doi: 10.1109/AISP.2012.6313817.
- [31] James R. Bainter, “Active Filter has stable notch and response can be regulated”, 1975
- [32] F. Wan, L. Wang, Q. Ji, y B. Ravelo, “Canonical transfer function of band-pass NGD circuit”, *IET Circuits, Devices & Systems*, vol. 13, núm. 2, pp. 125–130, mar. 2019, doi: 10.1049/iet-cds.2018.5214.
- [33] The PyWavelets Developers, “pywavelets API Reference”, <https://pywavelets.readthedocs.io/en/latest/ref/index.html>.
- [34] Telegram, “Telegram Bot API”. Consultado: el 12 de septiembre de 2023. [En línea]. Disponible en: <https://core.telegram.org/bots/api>

## ANEXOS

## ANEXO A. Modelo de Arquitectura del sistema

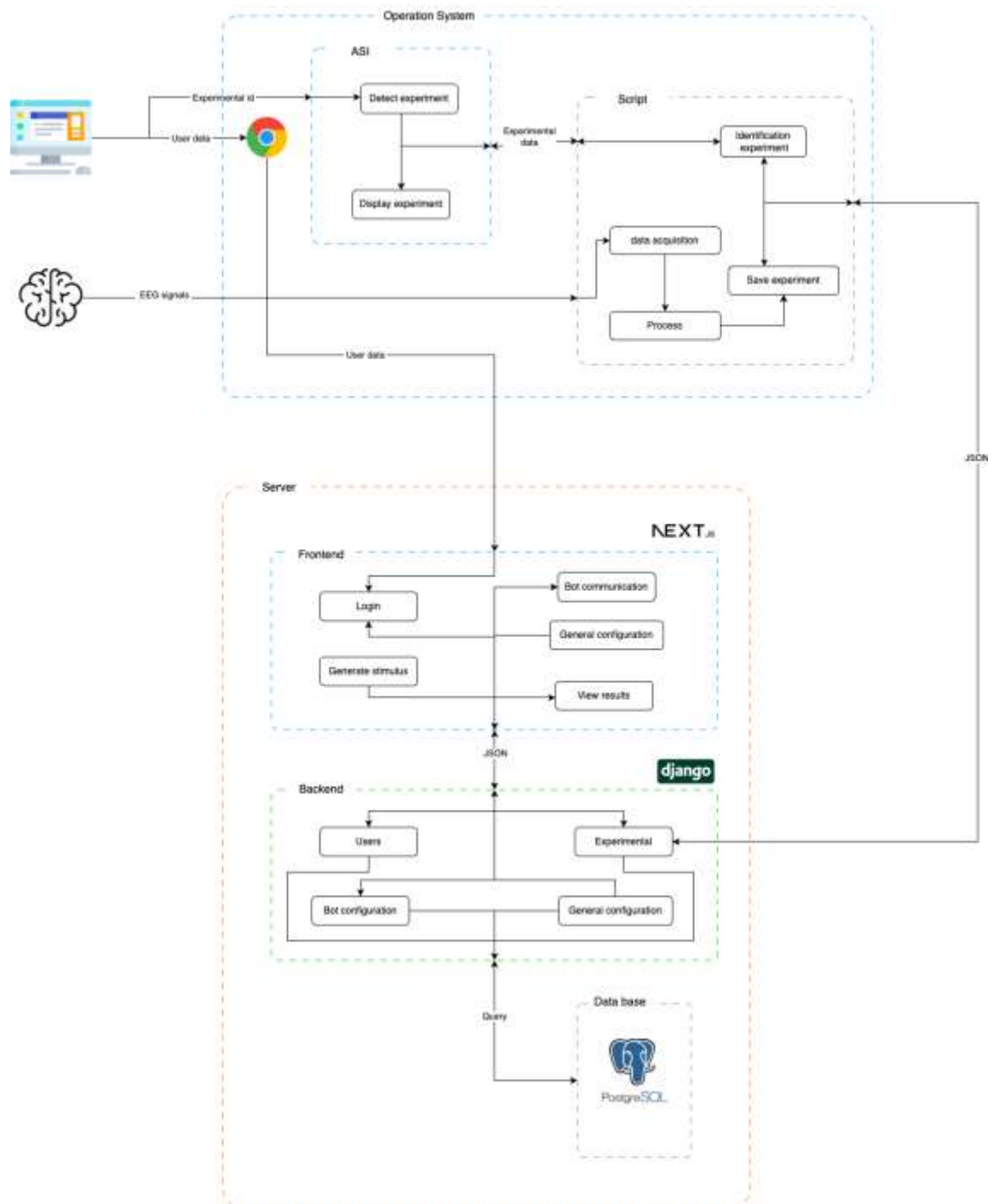


Figura 49. Arquitectura del sistema

## ANEXO B. Modelo Entidad – Relación de la base de datos

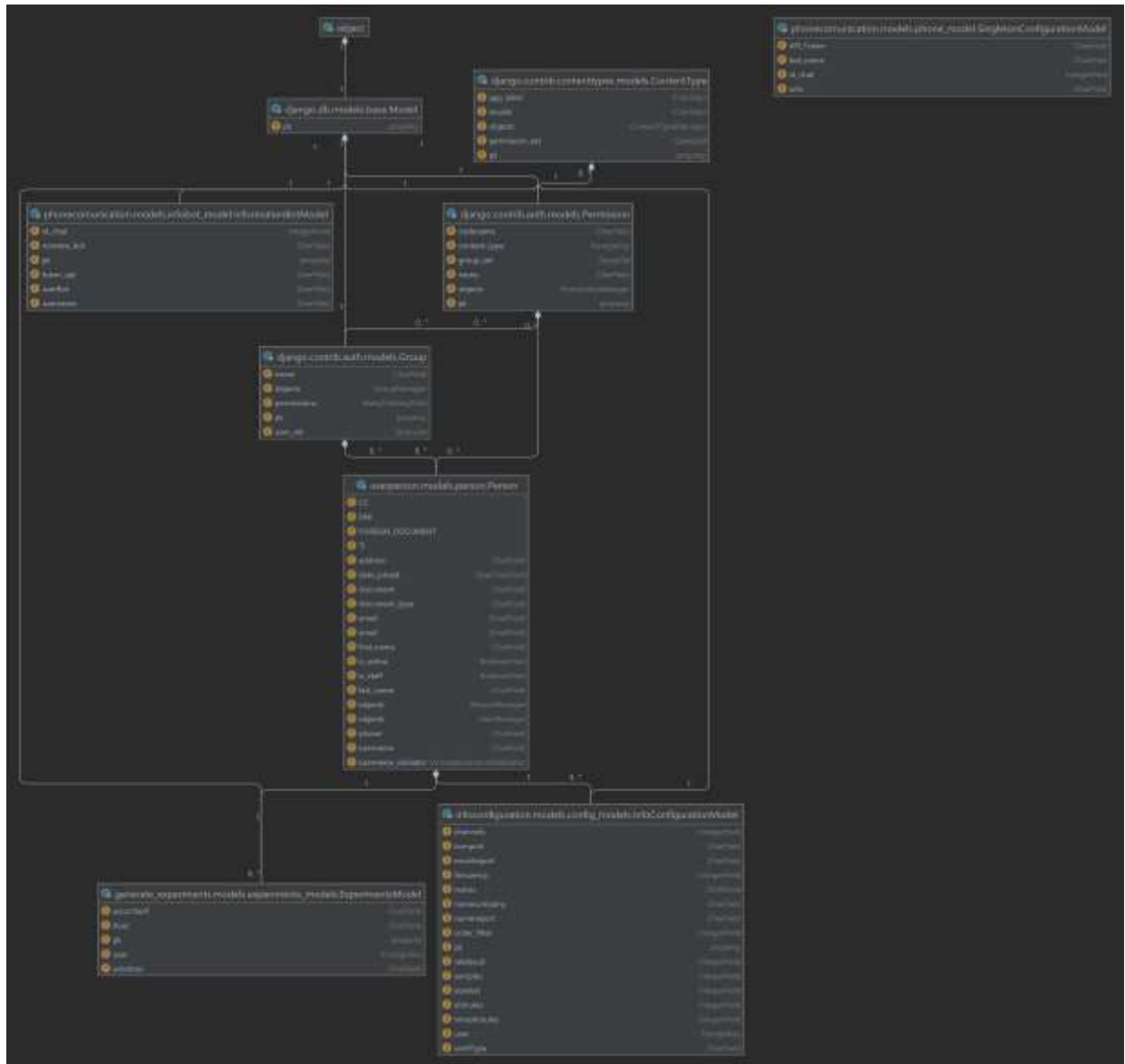


Figura 50. Modelo Entidad-Relación de la base de datos

## ANEXO C. Manual de Usuario Tarjeta por Integración *Cyton* de OpenBCI

La tarjeta Cyton es un dispositivo de adquisición de señales diseñado para capturar datos eléctricos generados por el cuerpo humano, como señales EEG (electroencefalograma), ECG (electrocardiograma), EMG (electromiografía) y otros tipos de señales bioeléctricas[28]. A continuación, se muestra los pasos para la conexión y uso de la tarjeta.



**Paso 1:** Preparación del Hardware. Asegúrese de tener la tarjeta Cyton en buenas condiciones y con todos los componentes necesarios.

**Paso 2:** Preparación de los Electrodo. Asegúrese de tener un conjunto de electrodos EEG compatibles. Estos electrodos generalmente tienen una forma circular y están hechos de materiales conductores.

**Paso 3:** Identificación de los pines de Conexión en la Tarjeta Cyton. Sujete la tarjeta Cyton por la parte frontal, lo que permitirá ver claramente los pines de conexión dispuestos en dos líneas, una superior y una inferior.

**Paso 4:** Conexión de los Electrodo. Comience conectando los electrodos en los pines de la tarjeta Cyton de izquierda a derecha, comenzando desde el pin NP1 y avanzando hasta llegar al pin BIAS, tal como se indica en la tarjeta. Asegúrese de que la conexión VCC del electrodo esté en el pin SRB y que la conexión a tierra esté en el pin GND.

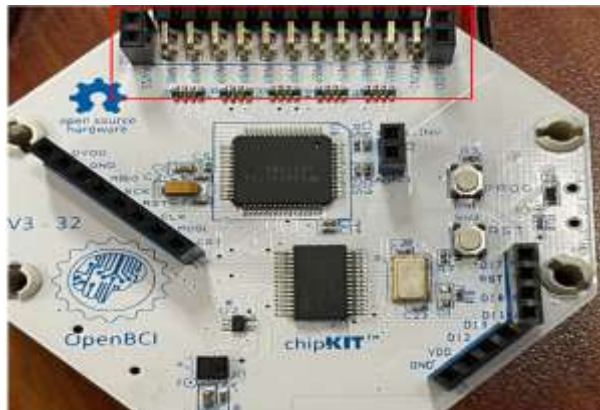


Figura 51. Tarjeta Cyton con especificación de pines por canal obtenida de [28]

Los electrodos se deben conectados a la tarjeta de la siguiente manera:

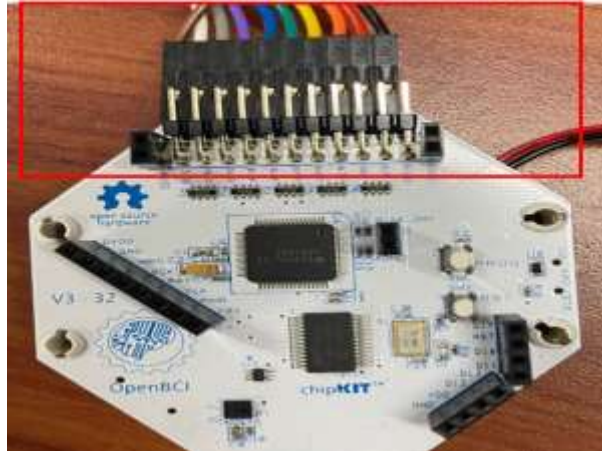


Figura 52. Conexión de electrodos a la tarjeta Cyton obtenido de [28]

Una vez tenga conectados ya se puede realizar la conexión de la alimentación, la tarjeta ya tiene dispuesto unos pines a los cuales se le puede conectar el módulo con pilas AA.

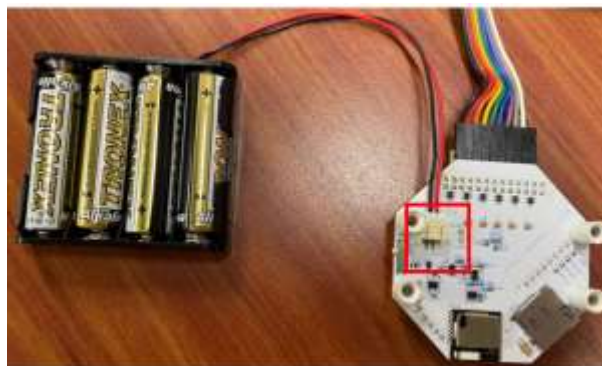


Figura 53. Conexión de fuente de alimentación a tarjeta Cyton obtenida de [28]

Con la tarjeta alimentada ya se encender y configurar para poderla conectar al dongle por lo cual se va a encontrar en la parte derecha vista la tarjeta desde el frontal un selector con las tres opciones distintas [28]:

- BLE activa un pull-up en el pin 4 de RFduino
- OFF desconecta la entrada de la batería
- La PC NO activa el pull-up en el pin 4 de RFduino

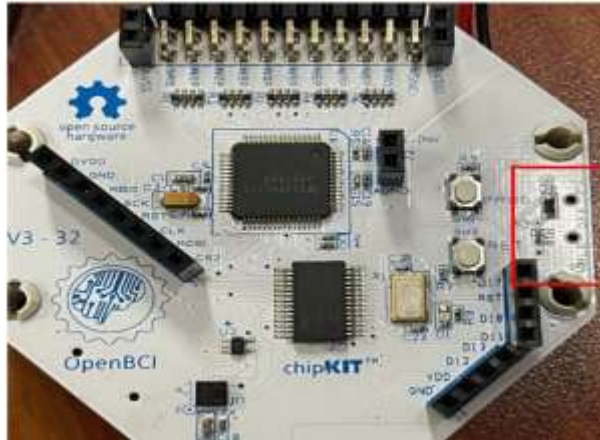


Figura 54. Indicación de pines de activación para tarjeta Cyton obtenida de [28]

### Configuración Hardware *Dongle*

El OpenBCI USB Dongle es utilizado para conectar su computadora a la placa Cyton[28].

**Paso 1:** Conexión Física. Asegúrese de que su computadora esté apagada. Inserte el OpenBCI USB *Dongle* en un puerto USB disponible de su computadora. Esto proporcionará la conexión física necesaria.

**Paso 2:** Encender el *Dongle*. En el *Dongle*, encontrará un interruptor deslizante con dos posiciones. El interruptor tiene dos lados, "GPIO6" y "RESET", que están marcados en la parte inferior. Asegúrese de que el interruptor esté en la posición "GPIO6". Esta es la configuración normal para el modo de operación estándar y permite la comunicación entre el *Dongle* y la placa Cyton.



Figura 55. Activación para funcionamiento dispositivo USB para comunicación con tarjeta Cyton obtenida de [28]

**Paso 3:** Verificación de la Conexión. Encienda su computadora y asegúrese de que el *Dongle* esté conectado correctamente a través del puerto USB. El *Dongle* debería estar listo para comunicarse con la placa Cyton.

**Paso 4:** Uso con la Placa Cyton. Ahora que el *Dongle* está configurado y conectado, puede utilizarlo para comunicarse con la placa Cyton. Asegúrese de que ambos dispositivos estén encendidos y configurados adecuadamente para la adquisición de datos.

### **Conexión de electrodos**

Para realizar la preparación de los electrodos y poder hacer el registro de datos siga los siguientes pasos:

**Paso 1:** Aplique gel conductivo o pasta conductora en cada uno de los electrodos. Esto ayuda a mejorar la conductividad eléctrica entre la piel y los electrodos para obtener señales más limpias.

**Paso 2:** Conecte los electrodos a los cables de extensión o cables adaptadores que sean compatibles con la tarjeta. Asegúrese de que los cables tengan el tipo de conector adecuado para la tarjeta.

**Paso 3:** Coloque los electrodos en la posición correcta en el cuero cabelludo según la configuración espacial deseada. De acuerdo a cada configuración se proporciona una ubicación estandarizada para cada electrodo y se utiliza ampliamente en la investigación EEG.

### **ANEXO D. Manual de Usuario para plataforma web y administración de usuarios**

Este manual le proporcionará información detallada sobre cómo utilizar esta interfaz, que permite a las personas expresar sus necesidades y opiniones utilizando actividades de deletreo. A lo largo de este manual, encontrará instrucciones paso a paso sobre cómo utilizar la interfaz BCI, qué funciones están disponibles y cómo aprovechar al máximo esta herramienta. Además, se proporcionarán detalles sobre la configuración y los procedimientos necesarios para una correcta comunicación y adquisición de señales EEG.

### ***Login***

Para ingresar, se requiere tener un usuario y contraseña que se generarán de manera personalizada dentro de la plataforma. En este módulo de inicio de sesión, los usuarios encontrarán un entorno seguro y protegido que garantiza la privacidad de la información. A través de esta interfaz, se podrá acceder a una serie de funcionalidades diseñadas para facilitar la comunicación y el uso de la plataforma.

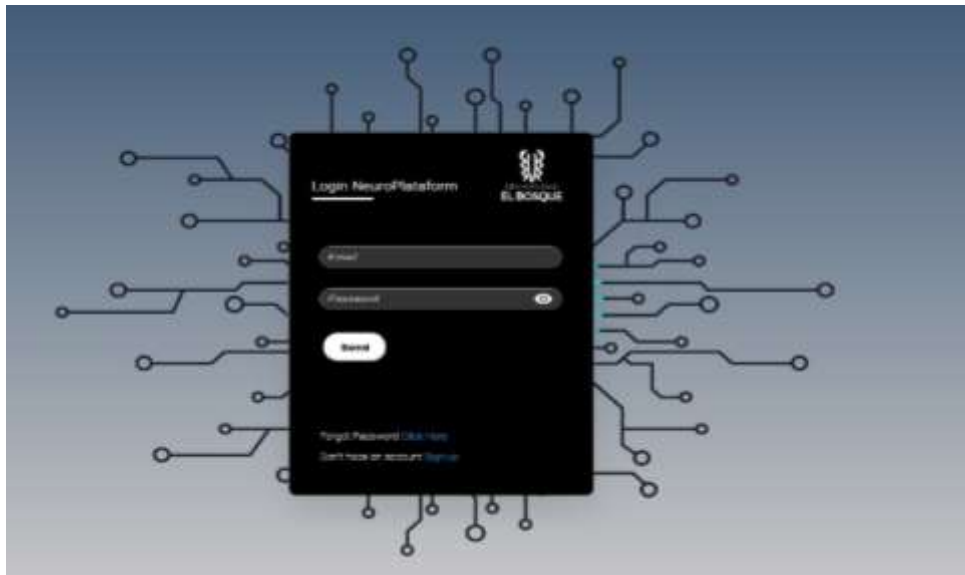


Figura 56. Vista de acceso tipo *Login*

### **Vista de inicio**

Esta vista es el primer punto de encuentro con la plataforma una vez que se ha completado el proceso de inicio de sesión. Aquí, los usuarios tendrán acceso a un menú desplegable ubicado en la parte izquierda de la ventana. Este menú ofrece una lista de opciones que corresponden a cada uno de los módulos disponibles en la plataforma. La "Vista de Inicio" es el punto de partida para navegar por la plataforma y acceder a las distintas funcionalidades que ofrece. A través de este menú, los usuarios pueden seleccionar el módulo específico que desean utilizar.

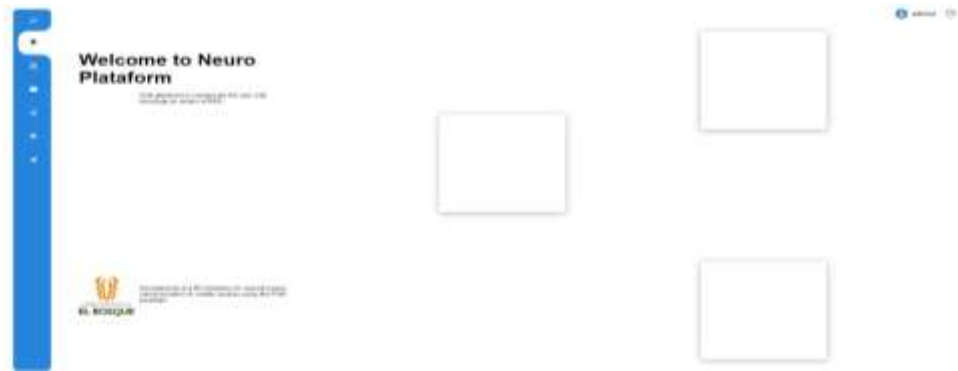


Figura 57. Vista principal de plataforma web

### Vista de experimentos

En este módulo, denominado "Vista de Experimentos", los usuarios tendrán la oportunidad de llevar a cabo una serie de experimentos de manera eficiente y organizada.

A continuación, se detallan los pasos en secuencia:

**Paso 1:** Acceder al Módulo "*Experiments*": Ingrese a la plataforma y navegue hacia el módulo denominado "Vista de Experimentos."

**Paso 2:** Visualizar la Guía del Sistema 10-10: Una vez dentro del módulo, se encontrará con una imagen que representa el Sistema 10-10, una guía visual para la colocación adecuada de los electrodos en el cuero cabelludo. Esta guía utiliza letras para identificar áreas específicas en el cuero cabelludo, como las zonas frontal (Fp y F), temporal (T), parietal (P), central (C), y occipital (O), junto con dos electrodos de referencia en las orejas (A).

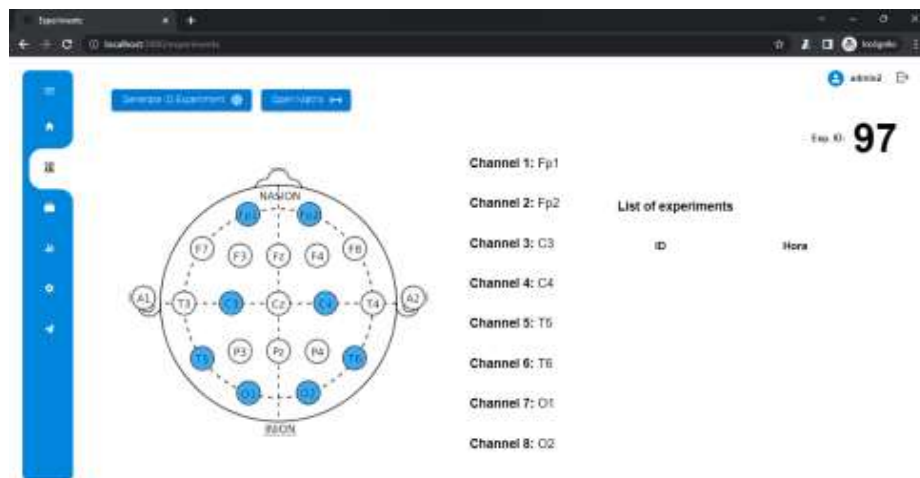


Figura 58. Vista de Experimentos de la plataforma web

**Paso 3:** Generar un ID de Experimento: Dentro del módulo de experimentos, encontrará un botón denominado "*Generate ID Experiment.*" Haga clic en este botón para generar un ID único para el experimento. Este ID se utilizará para identificar y seguir el experimento.

**Paso 4:** Ingresar el ID en la Interfaz Local: El ID generado debe ser ingresado en la interfaz local de la plataforma. Asegúrese de ingresar el ID correctamente para que el proceso sea exitoso.

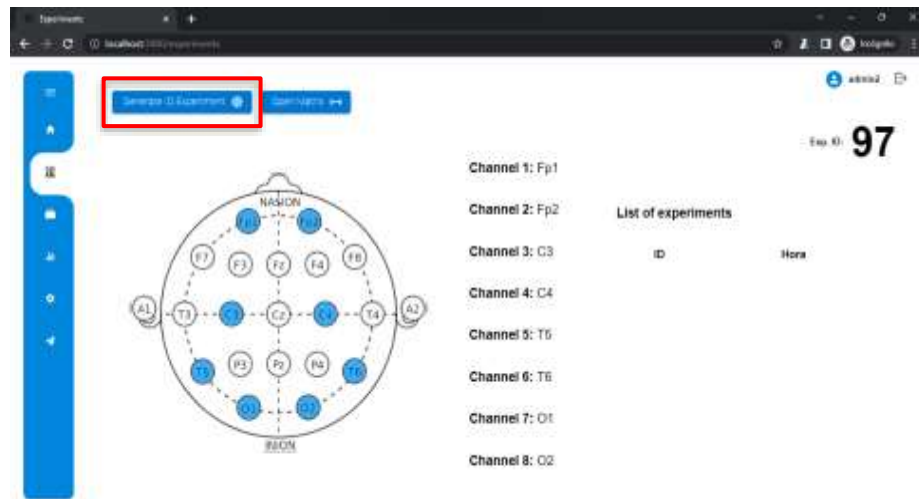


Figura 59. Especificación de botón para generación de ID de experimento

**Paso 5:** Esperar el Horario Programado: Una vez que se haya ingresado el ID en la interfaz local, espere hasta que llegue el horario programado para el experimento. La plataforma le indicará cuándo es el momento adecuado para comenzar.

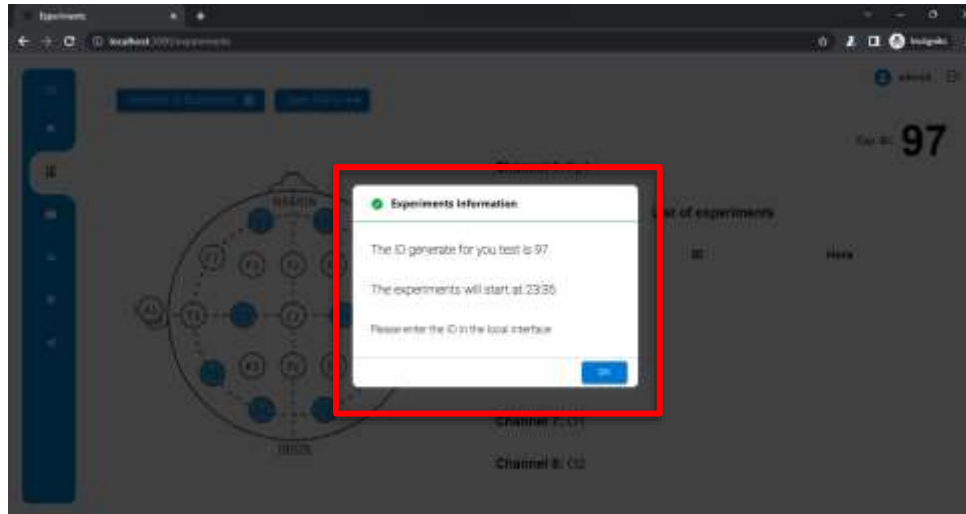


Figura 60. Alerta de ID y hora de inicio del experimento generado

**Paso 6:** Abrir la Matriz de Deletreo: Cuando llegue el horario programado, se habilitará un botón especial para abrir la matriz de deletreo. Haga clic en este botón para abrir una nueva ventana en el navegador y visualizar la matriz a pantalla completa.

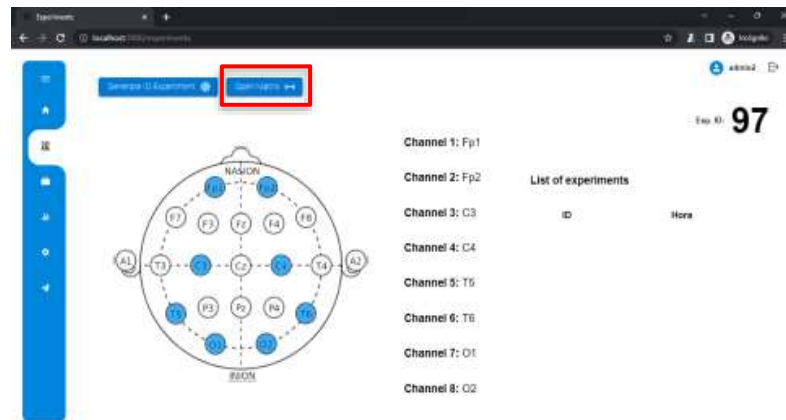


Figura 61. Indicación de botón para abrir vista de matriz

**Paso 7:** Inicio del Experimento: La interfaz mostrará un mensaje informativo que indicará que el experimento está a punto de comenzar. A partir de este momento, el experimento se iniciará y la matriz comenzará a resaltar filas y columnas, ejecutando los procesos de identificación.





Figura 62. Mensaje informativo en la vista matriz sobre hora de inicio

**Paso 8:** Realizar el Deletreo: Una vez que el experimento ha comenzado y la matriz está en funcionamiento, el paciente estará listo para realizar el proceso de deletreo. Siga las instrucciones y realice el deletreo de acuerdo con las indicaciones.



Figura 63. Parpadeo de filas y columnas en la vista de matriz

Una vez que el proceso de deletreo ha concluido, tanto el paciente como el administrador podrán acceder a una sección de estadísticas que proporciona información valiosa sobre el rendimiento y los resultados del experimento. A continuación, se describen los pasos para acceder a esta sección y comprender las estadísticas:

**Paso 9:** Acceder a la Pestaña de Estadísticas: Después de que el proceso de deletreo haya finalizado, busque y seleccione la opción que le permita acceder a la pestaña de estadísticas en la plataforma.

**Paso 10:** Visualizar la Información de Rendimiento: En la pestaña de estadísticas, encontrará información relevante sobre el rendimiento durante el experimento. A continuación, se detallan los elementos clave que se presentarán:

- **Contador de Épocas:** En esta sección se puede visualizar y conteo total de la cantidad de épocas por cada filas y columnas en función de la cantidad de secuencias generadas.

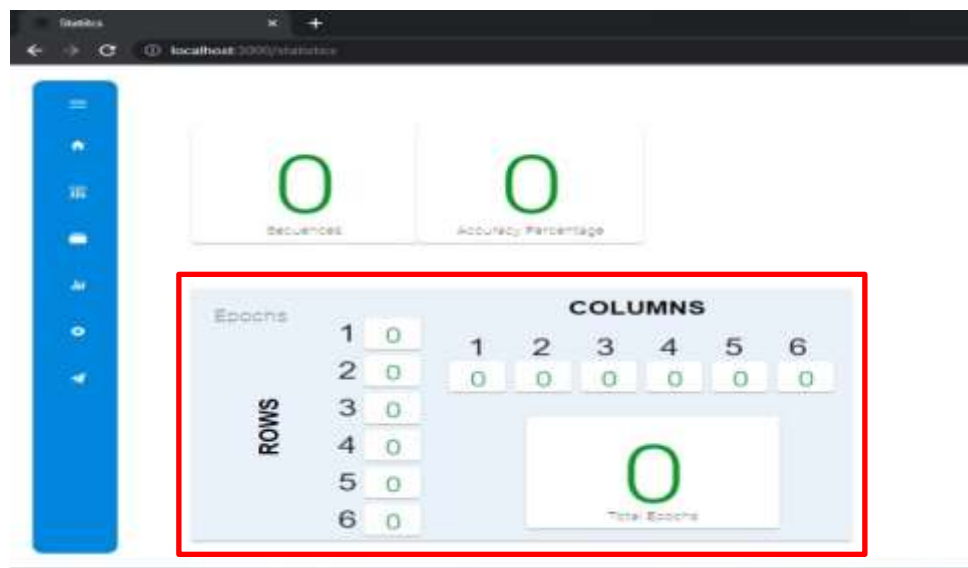


Figura 64. Visualización de cantidad de épocas por sección y secuencias

- **Contador de Secuencias:** Esta métrica registra la cantidad de secuencias completadas durante el proceso de deletreo. El contador de secuencias permite evaluar la fluidez y el progreso en la comunicación a lo largo del experimento.
- **Indicador de Precisión:** En esta sección, encontrará un indicador en porcentaje que representa la precisión del deletreo. Este indicador refleja la exactitud de las selecciones realizadas durante el proceso de deletreo, lo que resulta fundamental para evaluar la eficacia de la comunicación a través de la interfaz cerebro-ordenador.

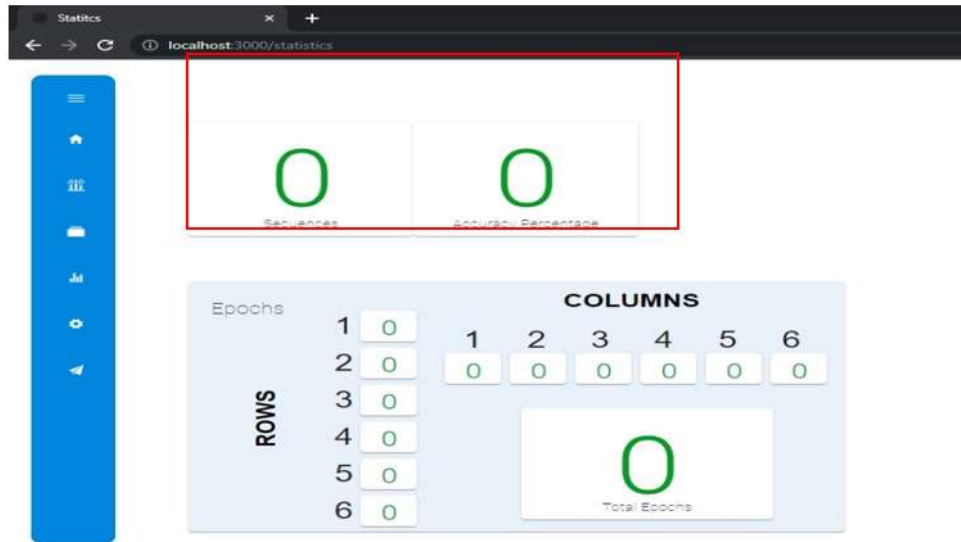


Figura 65. Visualización de cantidad de secuencias y porcentaje de precisión

- **Cantidad Total de Palabras Deletreadas:** Esta métrica muestra la cantidad total de palabras que se deletrearon durante la sesión. Proporciona una medida importante para evaluar la productividad y el éxito de la comunicación durante el experimento.

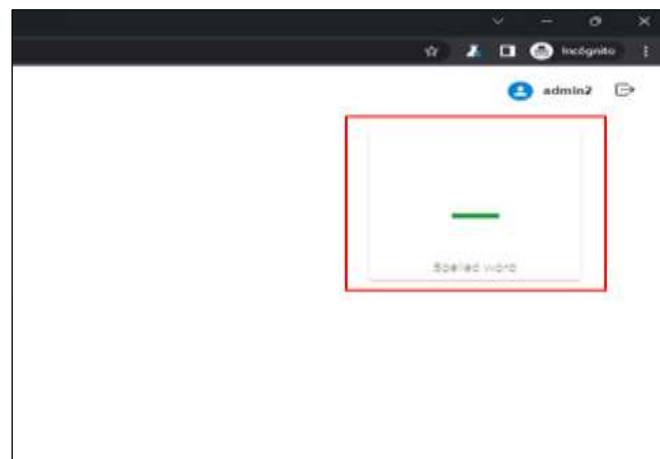


Figura 66. Visualización del carácter detectada

### Vista de configuración

La "Ventana de Configuración" es un componente clave de la plataforma que permite a los usuarios personalizar y ajustar diversos aspectos según sus necesidades. A continuación, se describen los pasos para configurar esta ventana de configuración de manera efectiva:

**Paso 1:** Acceder a la Ventana de Configuración: Inicie sesión en la plataforma utilizando sus credenciales. Una vez dentro de su cuenta, busque y seleccione la opción que le permita acceder a la "Vista de Configuración". Esto generalmente se encuentra en el menú de configuración o ajustes.

**Paso 2:** Configuración de Paradigma y Canales: En la primera subsección, relacionada con la configuración de paradigma y canales, encontrará opciones para modificar los canales utilizados en la plataforma. Aquí, puede personalizar la configuración de los canales según sus necesidades. Si desea modificar la cantidad de canales a utilizar, realice los ajustes necesarios.

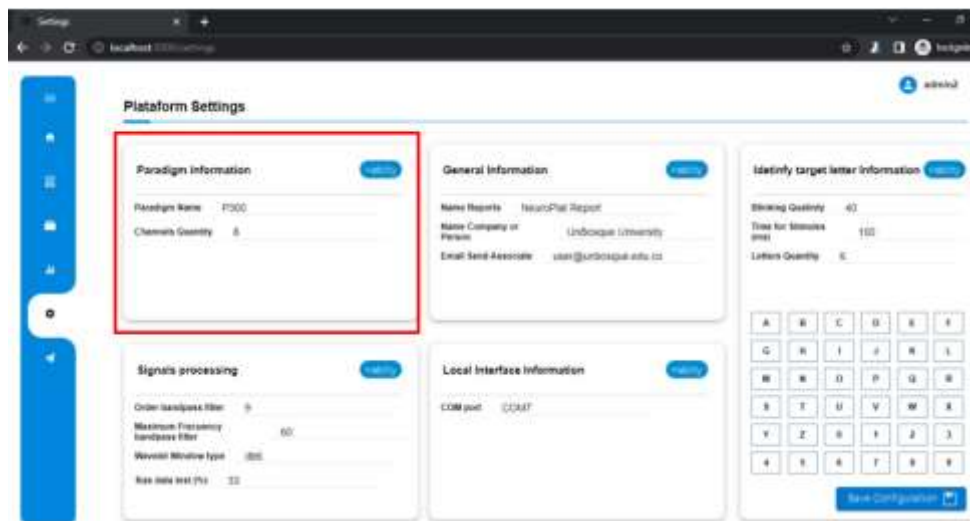


Figura 67. Vista de configuración, sección de información de paradigma

**Paso 3:** Información General: En esta subsección, puede configurar detalles generales relacionados con la plataforma. Esto incluye editar el nombre asociado a los informes generados por la plataforma, así como otros detalles generales de la entidad o usuario que utiliza la plataforma. Haga clic en los campos correspondientes para realizar las ediciones necesarias.

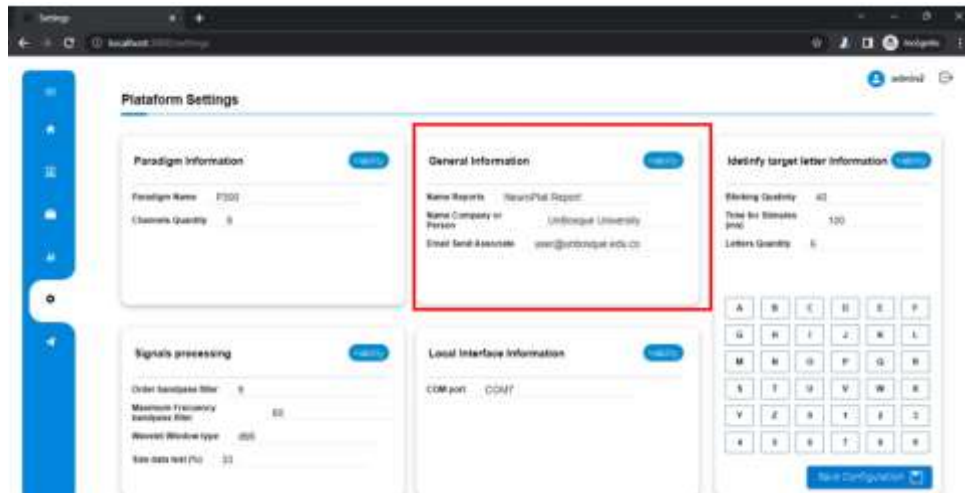


Figura 68. Vista de configuración, sección de información general

**Paso 4:** Configuración de Procesamiento de Señal: Diríjase a esta subsección para realizar configuraciones cruciales relacionadas con el procesamiento de señales. Aquí, puede definir el orden y la frecuencia de corte del filtro pasa banda. Recuerde que la frecuencia de corte máxima depende de la frecuencia de muestreo utilizada por la tarjeta de adquisición, según el teorema de Nyquist. Además, seleccione el tipo de familia wavelet que se utilizará para la transformada de wavelet. Configure el porcentaje de muestras a utilizar por el clasificador para pruebas

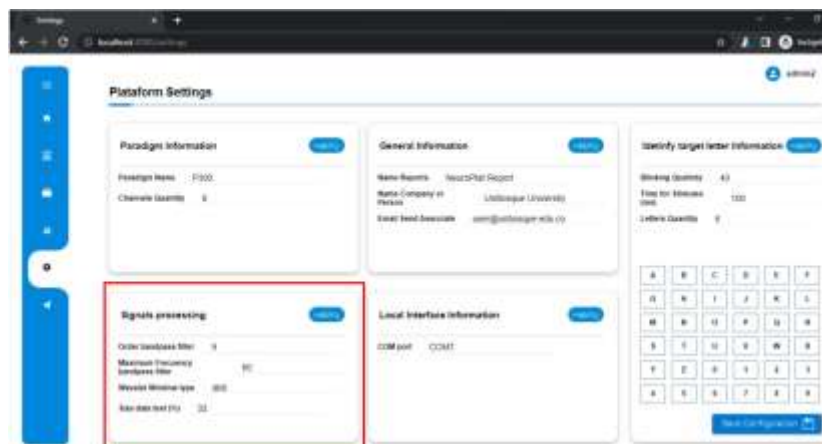


Figura 69. Vista de configuración, sección de procesamiento de señales

**Paso 5:** Información Local y Configuración del Puerto: En esta subsección, se proporciona información sobre la configuración del puerto COM al que está conectada la tarjeta de

adquisición de datos. Verifique y configure esta información para garantizar una comunicación efectiva y una sincronización precisa de datos entre la plataforma y los dispositivos de adquisición.

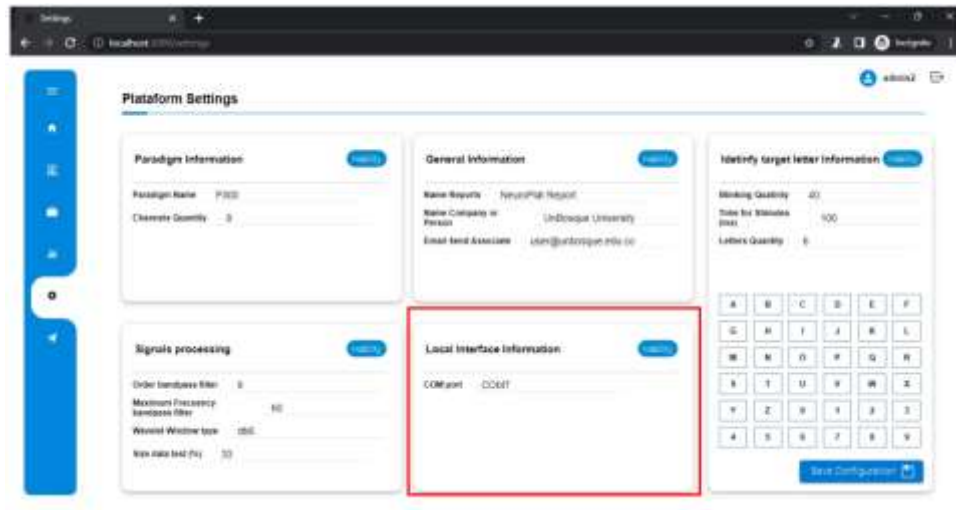


Figura 70. Vista de configuración, sección de información de interfaz local

**Paso 6:** Edición de Matriz de Identificación de Letra Target: Finalmente, dentro de esta subsección, puede realizar configuraciones relacionadas con la identificación de la letra objetivo. Defina la cantidad de parpadeos, el tiempo entre parpadeos y la cantidad de letras que se deben deletrear. Además, observe las casillas de visualización de la matriz de estímulos, donde puede ingresar hasta 4 caracteres por casilla. Haga los ajustes necesarios para adaptar esta configuración a los requerimientos de su experimento.

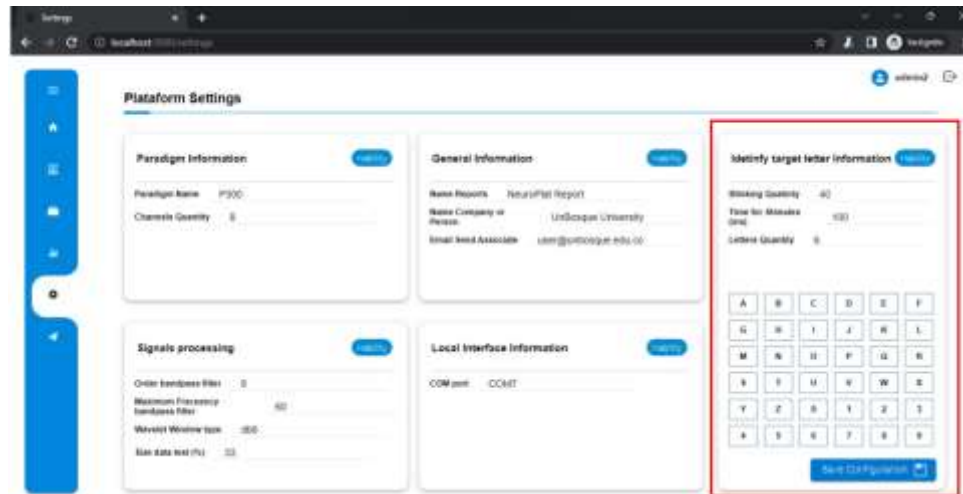


Figura 71. Vista de configuración, sección de información de configuración de la matriz

### Ventana de comunicación

La "Vista de Comunicación" es una parte fundamental de la plataforma que permite la comunicación entre la plataforma web y un bot en la aplicación de Telegram. A continuación, se detallan los pasos para configurar esta vista:

**Paso 1:** Acceder a la Vista de Comunicación: Inicie sesión en la plataforma web y asegúrese de haber ingresado a su cuenta. Busque y seleccione la opción que le permita acceder a la "Vista de Comunicación". Esto suele encontrarse en un menú de configuración o ajustes de la plataforma.

**Paso 2:** Configuración de Información General para la Aplicación de Telegram (*Information General to Telegram Application*):

- Dentro de la vista de comunicación, encontrará una sección destinada a la configuración de la información general para la aplicación de Telegram.
- Ingrese el nombre del bot de Telegram que ha creado y el API Token proporcionado por la aplicación de Telegram durante la creación del bot.
- Haga clic en el botón "Probe Connection" para que la plataforma envíe esta información al Backend que utiliza la API de Telegram.

- La plataforma verificará la correcta comunicación con la API de Telegram y obtendrá el ID del chat y el propietario que generó el bot. Estos datos se mostrarán en los componentes inferiores de la sección.
- Guarde los resultados de la conexión en la base de datos del sistema para su uso posterior.

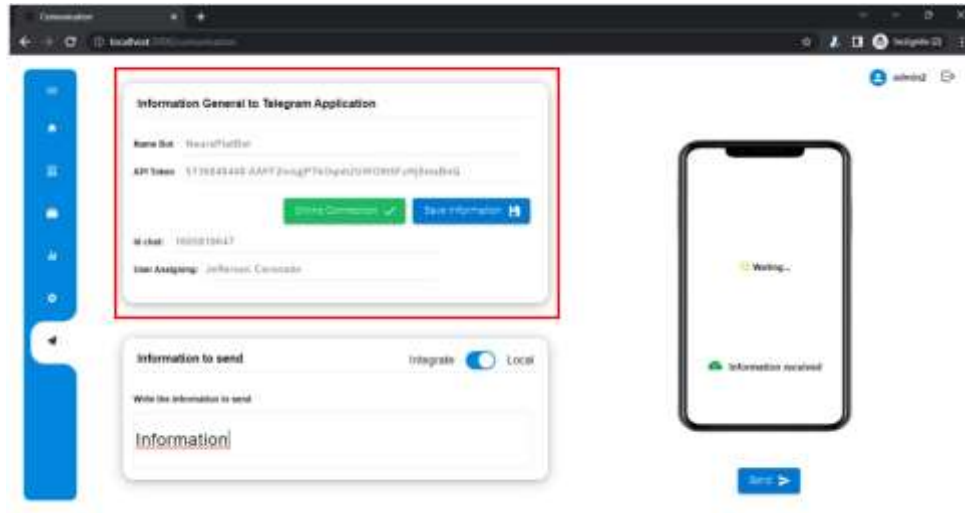


Figura 72. Vista de comunicación con el dispositivo móvil con configuración general del bot

**Paso 3:** Configuración de Información a Enviar (*Information to Send*): En la misma vista de comunicación, existe una sección para configurar la información que se enviará al bot de Telegram.

- La configuración base es "*Integrate*", lo que significa que el sistema enviará automáticamente información del carácter identificado por la vista de matriz al bot de Telegram.
- Si desea cambiar la configuración a "*Local*", puede hacerlo para permitir el envío de información escrita por el usuario.
- La modificación de esta subsección en conexión local se visualiza en la plataforma.



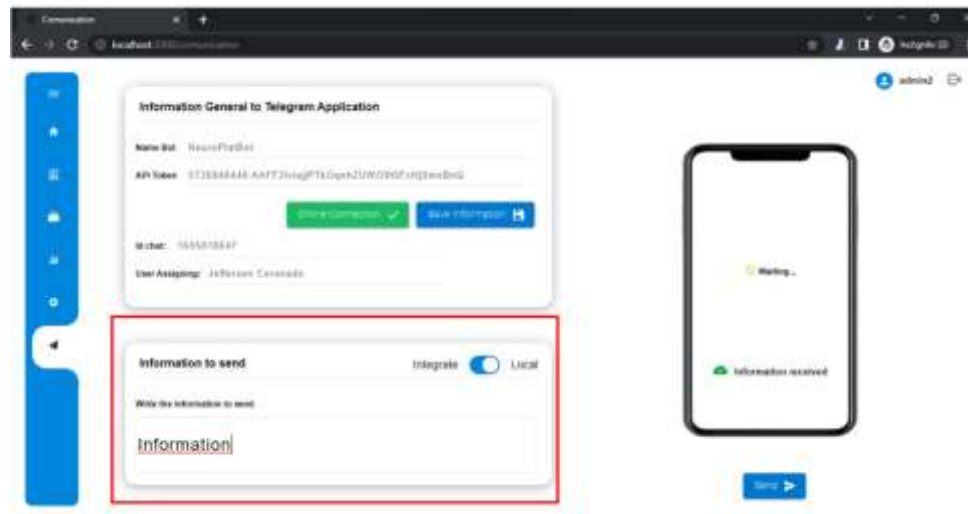


Figura 73. Envío local o integrado de información al dispositivo móvil

**Paso 4:** Activación del Bot de Telegram: Para establecer la conexión con el bot de Telegram, es necesario iniciar el bot por medio del comando /start en la aplicación de Telegram. Esto activará la comunicación del chat por medio de la API correspondiente. Consulte la documentación en el anexo D para obtener detalles sobre cómo ejecutar esta acción.

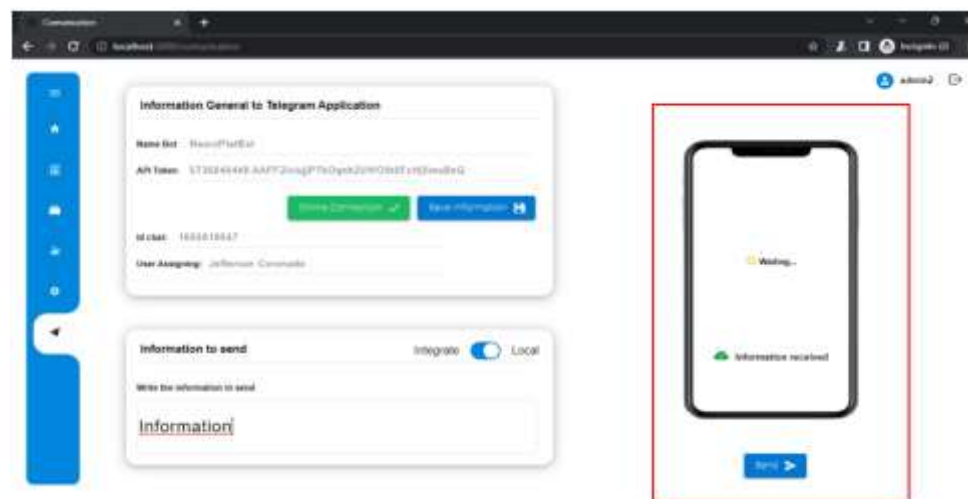


Figura 74. Visualización de estado de comunicación

### Conexión de API o *Endpoints de Backend*

La plataforma facilita la conexión de la API mediante el uso de *Swagger*, una herramienta que permite encontrar y visualizar los endpoints necesarios para establecer una conexión efectiva entre la plataforma y el *bot* de Telegram. Aquí, se describen dos métodos fundamentales para llevar a cabo esta conexión.

La utilización de *Swagger* simplifica el proceso de conexión de la API al proporcionar una interfaz clara y accesible para visualizar y configurar los endpoints necesarios. Esta funcionalidad es esencial para garantizar que la plataforma esté debidamente sincronizada con el *bot* de Telegram, lo que facilita la comunicación efectiva y el flujo de información entre ambas partes.



Figura 75. Visualización de documentación de API en el backend

### Administración Django

#### Login

La plataforma proporciona un panel de administración de usuarios a través de Django Rest Framework, lo que permite el control total sobre el backend, incluyendo la creación, actualización, modificación y eliminación de usuarios para el funcionamiento del sistema. Para acceder a esta plataforma, siga los siguientes pasos:

**Paso 1:** Abra un navegador web y diríjase a la URL del backend de la plataforma. Esto generalmente se hace ingresando la dirección URL en la barra de direcciones del navegador.

**Paso 2:** Una vez en la página del backend, agregue "/admin" al final de la URL. Esto lo llevará a la plataforma de administración de Django.

**Paso 3:** En la página de inicio de sesión, se le pedirá que ingrese sus credenciales de acceso. Esto incluye un "nombre de usuario" y una "contraseña".

Las credenciales de acceso corresponden a las creadas como superusuario durante el proceso de despliegue del backend, como se describe en el manual de usuario para despliegue local.

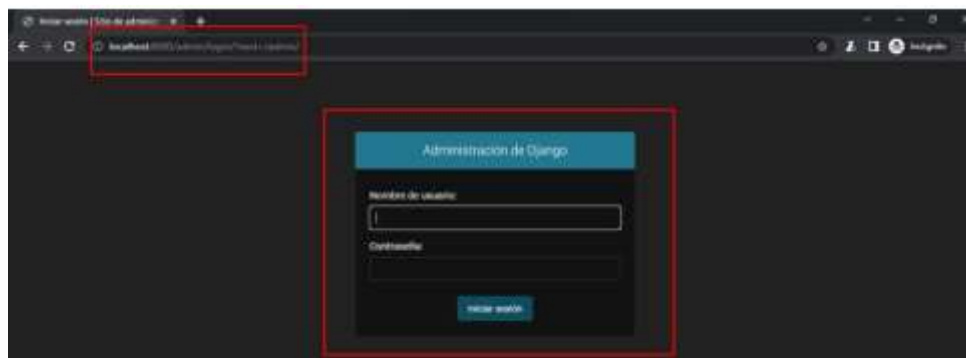


Figura 76. Plataforma de administración de usuarios

Una vez que haya ingresado las credenciales correctas, accederá a la interfaz de administración de Django. Esta interfaz proporciona un control completo sobre el backend, lo que permite realizar diversas acciones relacionadas con la gestión de usuarios y configuraciones avanzadas dentro de la plataforma.

### Sitio administrativo

El "Sitio Administrativo" de la plataforma brinda a los usuarios con privilegios avanzados un entorno completo para gestionar y controlar aspectos clave de la aplicación. Para acceder a esta sección, el usuario debe seguir estos pasos:

**Paso 1:** Iniciar Sesión: El usuario debe iniciar sesión en la plataforma utilizando sus credenciales de acceso.

**Paso 2:** Acceder al Sitio Administrativo: Una vez dentro de la plataforma, el usuario debe buscar la opción que le permita acceder al "Sitio Administrativo". Por lo general, encontrará un enlace o botón etiquetado como "Sitio Administrativo" o "Admin". Al hacer clic en esta opción, accederá al entorno de administración.

**Paso 3:** Gestión de Grupos: En esta sección, el usuario tiene la capacidad de agregar nuevos grupos o editar los grupos existentes. Los grupos se utilizan para organizar y categorizar usuarios, facilitando así la asignación de permisos y el control de acceso a funciones específicas de la plataforma.

- Para agregar un nuevo grupo, el usuario debe buscar la opción correspondiente y seguir las instrucciones para definir el nombre y configurar los permisos.
- Para editar un grupo existente, el usuario puede seleccionar el grupo deseado de la lista y realizar las ediciones necesarias en su configuración.

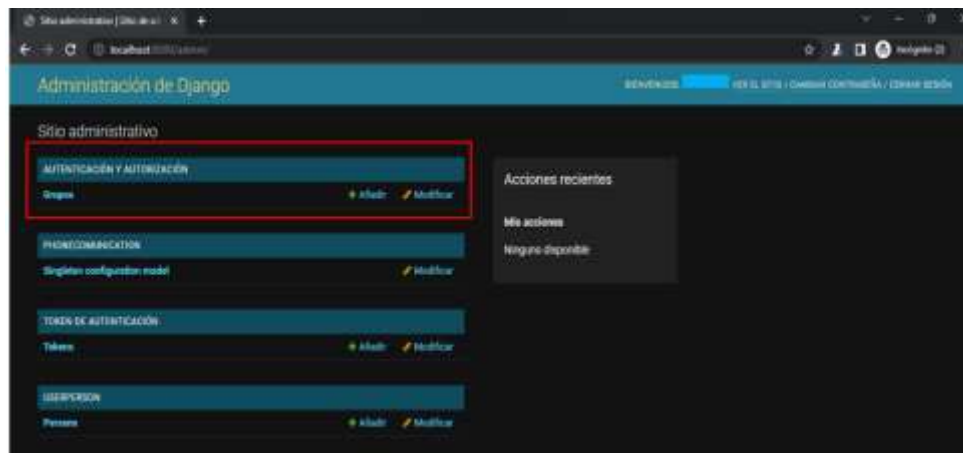


Figura 77. Administración de grupos de personas

**Paso 4:** Configuración de Comunicación con el Teléfono: En esta sección, el usuario puede personalizar y configurar la comunicación de la plataforma con dispositivos móviles, como teléfonos y *tablets*. Aquí, se pueden definir parámetros de comunicación, configurar notificaciones y asegurarse de que la plataforma esté lista para interactuar de manera eficiente con dispositivos móviles.

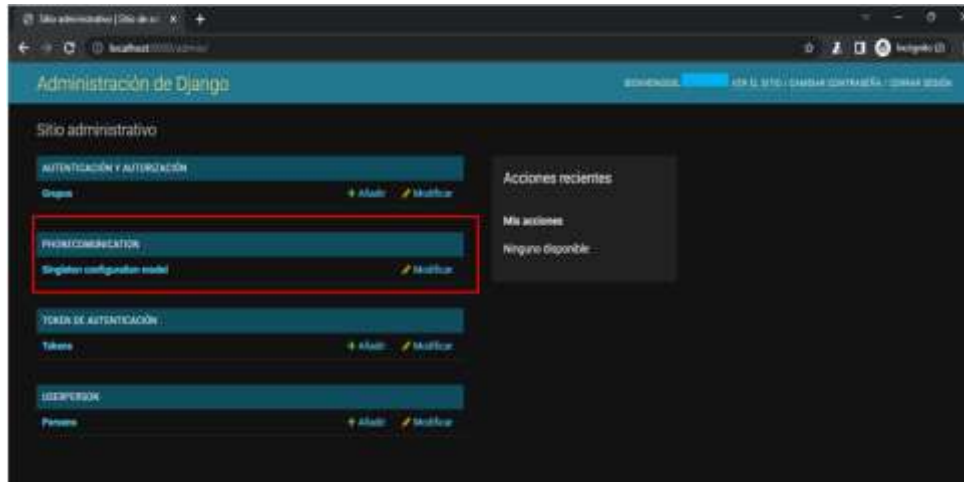


Figura 78. Administración de modelos para configuración de bot

- El usuario debe explorar las opciones disponibles para ajustar la configuración de comunicación según sus necesidades.

**Paso 5:** Gestión de Tokens: En esta sección, el usuario puede agregar o editar tokens. Los tokens son esenciales para garantizar la autenticación segura y el acceso a la plataforma. Se pueden gestionar y asignar tokens a usuarios específicos o dispositivos para garantizar la seguridad y la autorización adecuada.

- El usuario debe utilizar las opciones disponibles para crear nuevos tokens o editar los existentes según sea necesario.

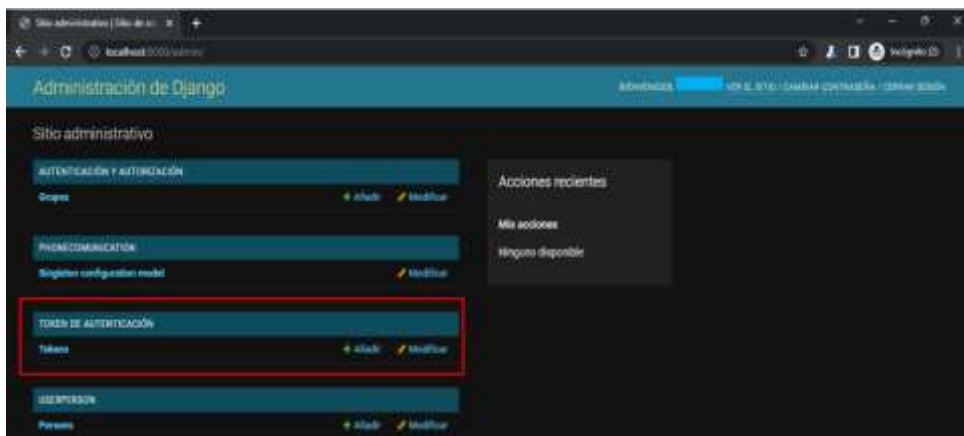


Figura 79. Administración de tokens de acceso

**Paso 6:** Gestión de Personas: Aquí, el usuario puede agregar o editar información sobre las personas registradas en la plataforma. Esto incluye detalles de contacto, roles y permisos, lo que permite una administración completa de los usuarios.

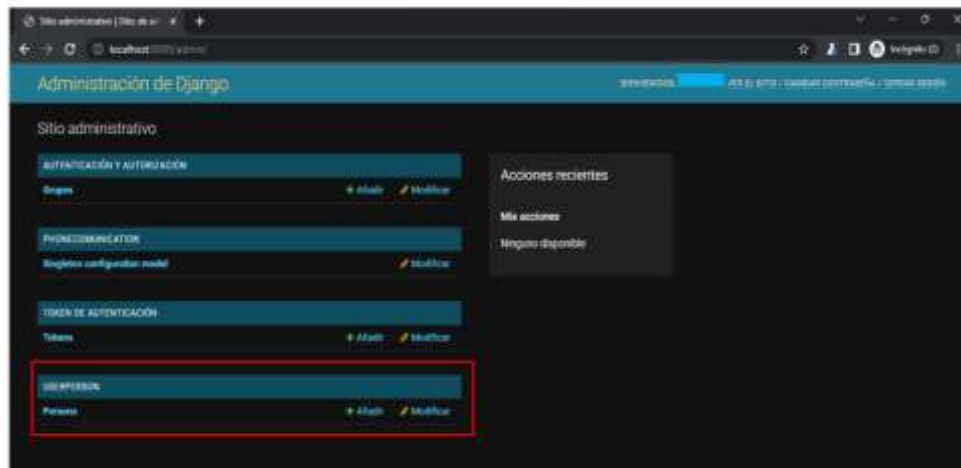


Figura 80. Administración de usuarios ya creados

Cuando un usuario con privilegios avanzados accede al "Sitio Administrativo" y selecciona la opción "Personas", se abre una sección que permite la gestión y visualización de los usuarios registrados en la plataforma. A continuación, se detallan los pasos para gestionar personas en esta sección:

**Paso 1:** Iniciar Sesión y Acceder al Sitio Administrativo: El usuario con privilegios avanzados debe iniciar sesión en la plataforma utilizando sus credenciales de acceso. Luego, debe acceder al "Sitio Administrativo" siguiendo las instrucciones proporcionadas previamente.

**Paso 2:** Acceder a la Sección de Personas: Una vez dentro del "Sitio Administrativo", el usuario debe buscar y seleccionar la opción "Personas". Esta acción abrirá una tabla que presenta una visión general de todos los usuarios registrados en la plataforma.

**Paso 3:** Visualizar la Tabla de Usuarios: La tabla de usuarios proporciona información clave sobre los usuarios registrados, lo que facilita su identificación y gestión. A continuación, se describen las columnas de la tabla y la información que contienen:

- **Nombre de Usuario:** En esta columna, se muestra el nombre de usuario único de cada persona registrada en la plataforma. El nombre de usuario es un identificador único utilizado para acceder a la plataforma.
- **Correo Registrado:** Esta columna muestra la dirección de correo electrónico asociada a cada usuario. El correo electrónico es esencial para la comunicación y la autenticación en la plataforma.
- **Tipo de Documento:** Aquí se indica el tipo de documento utilizado para la identificación de cada usuario. Esto puede incluir documentos como pasaportes, cédulas de identidad, licencias de conducir, entre otros.
- **Estado:** En esta columna, se muestra el estado actual de cada usuario, lo que puede indicar si están activos o inactivos en la plataforma. Este estado puede estar relacionado con el acceso o la capacidad de utilizar ciertas funciones.
- **Staff:** Esta columna indica si un usuario tiene el estado de "Staff". Los usuarios con este estado pueden tener privilegios adicionales o acceso a funciones específicas dentro de la plataforma.

**Paso 4:** Editar Usuario: Si se desea editar la información de un usuario específico, el usuario con privilegios avanzados puede hacer clic en el botón de "Editar" que se encuentra junto a la entrada del usuario correspondiente en la tabla. Esto abrirá una interfaz de edición que permite realizar cambios en la información del usuario, como su nombre de usuario, correo electrónico, tipo de documento, estado y estado "Staff".

**Paso 5:** Guardar Cambios: Después de realizar las ediciones necesarias, el usuario con privilegios avanzados debe asegurarse de guardar los cambios haciendo clic en el botón "Guardar" o "Guardar Cambios", según la interfaz de edición.

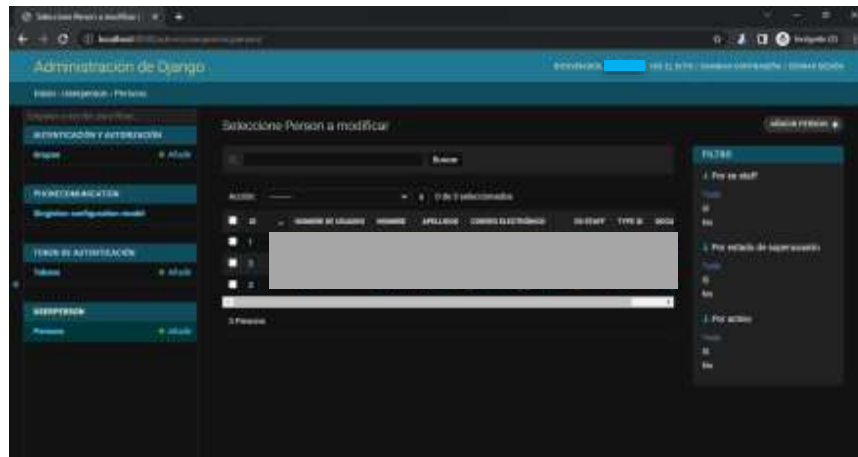


Figura 81. Panel de modificación y actualización de usuarios

Cuando se selecciona la opción de "Registrar Usuario," se accede a un formulario diseñado para recopilar información básica y necesaria sobre el nuevo usuario. A continuación, se describen los campos que deben completarse en este formulario:

- **Nombre de Usuario:** En este campo, se debe ingresar un nombre de usuario único que identificará al nuevo usuario en la plataforma. El nombre de usuario es esencial para el acceso y la autenticación.
- **Contraseña:** Aquí se debe definir una contraseña segura para el nuevo usuario. La contraseña debe cumplir con ciertos requisitos de seguridad, como tener una longitud mínima y contener caracteres especiales, letras mayúsculas y minúsculas, y números.
- **Tipo de Documento:** En este campo, se debe seleccionar el tipo de documento que el nuevo usuario utilizará para la identificación. Los tipos de documentos comunes pueden incluir pasaportes, cédulas de identidad, licencias de conducir, entre otros.
- **Número de Documento:** En este campo, se debe ingresar el número de documento correspondiente al tipo seleccionado. Este número es importante para la identificación precisa del usuario.



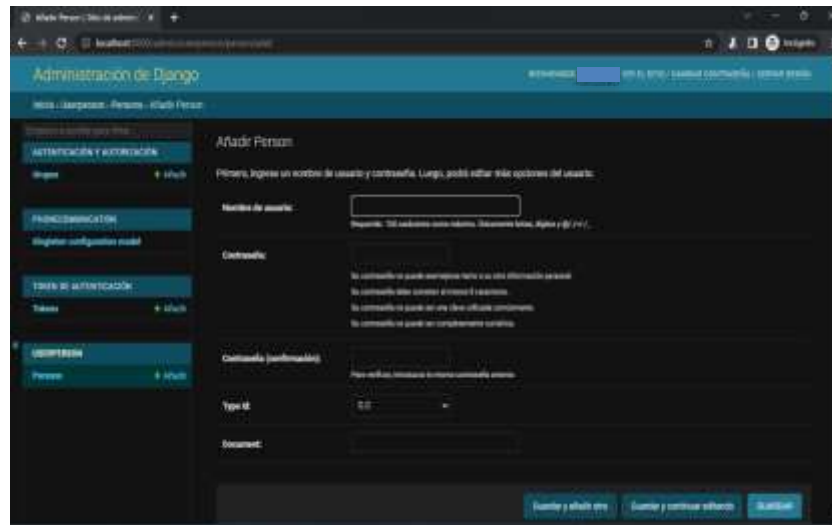
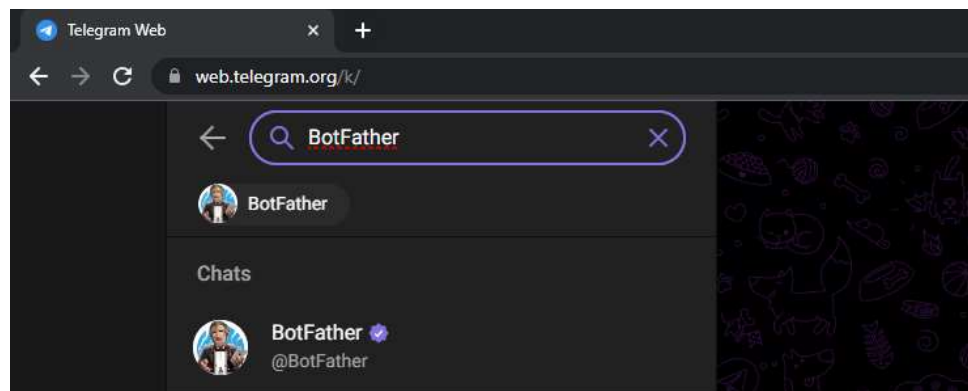


Figura 82. Panel de creación de usuarios

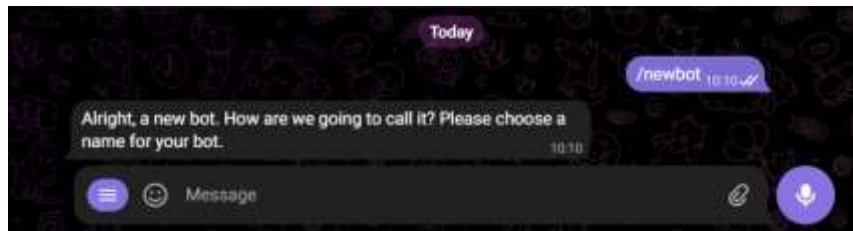
### Creacion de *bot* utilizando *BotFather* de Telegram

Para generar un bot que permita ser integrado a la plataforma web y se realice la comunicación correspondiente de acuerdo con la clasificación puede realizarse atreves del servicio *BotFather* de Telegram el cual es autónomo y cualquier usuario puede utilizarlo.

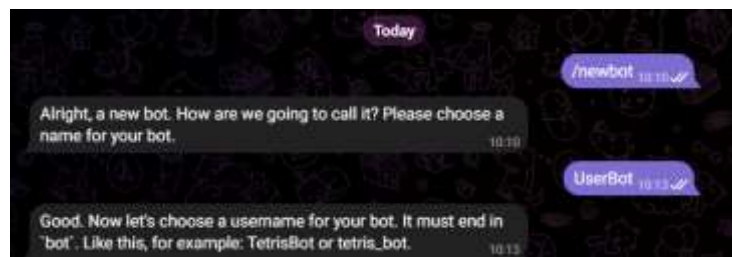
**Paso 1:** En la aplicación, en el icono de buscar, escribir “*BotFather*” e ingresar al chat expuesto.

Figura 83. Búsqueda de *bot* para creación de otros *bots* en Telegram

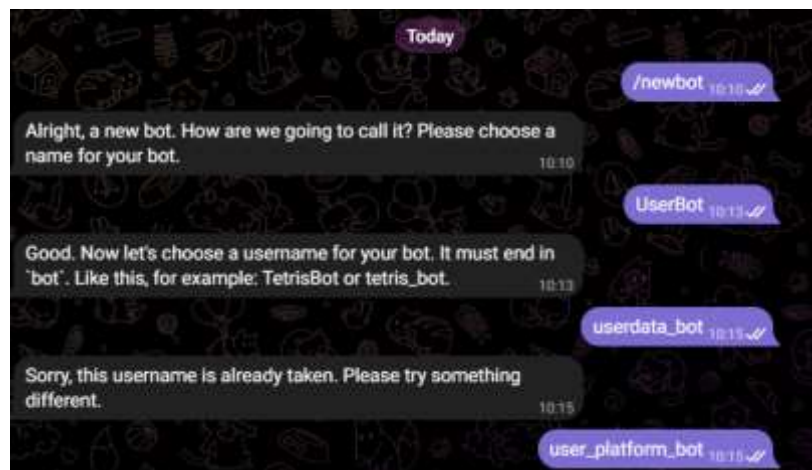
**Paso 2:** Escribir el comando `/newbot`. Con el cual generara un mensaje informativo automático por parte del chat.

Figura 84. Comando para creación de *nuevo bot*

**Paso 3:** Indicar un nombre para el bot. Este debe ser sin espacios y utilizando palabras asociados

Figura 85. Comando para asignación de nuevo nombre al *bot*

**Paso 4:** Indicar un *username* para el chat. Debe finalizaren con la palabra *bot*. En caso de que el *username* ya esté en uso, el chat indicara que debe seleccionar otro.

Figura 86. Asignación de username del nuevo *bot*

**Paso 5:** El sistema entregara el *username*, url de acceso y HTTP API Token del bot creado, esta información es la que se debe indicar en la vista *Communication* para verificar conexión y envío de datos.

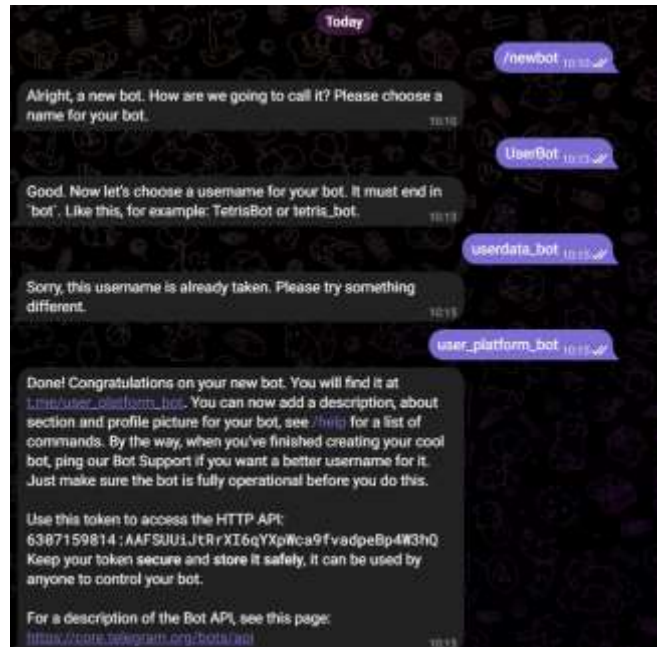


Figura 87. Información de creación del nuevo *bot*

En caso de requerir eliminar un bot previamente creado se debe utilizar el comando */deletebot* e indicar el nombre del bot que desea eliminar.

## ANEXO E. Manual de Usuario interfaz de adquisición de señales

El siguiente manual presenta los pasos para el manejo de la interfaz de adquisición, la cual es encargada de realizar la comunicación con la tarjeta de adquisición y realizar el procesamiento respectivo de la aplicación.

**Paso 1:** Ejecutar la aplicación de escritorio que permite acceder al script de acceso asociado.

**Paso 2:** Generar un ID de experimento en la plataforma web.

**Paso 3:** Ingresar el ID indicado en el campo “*ID Experiment*”

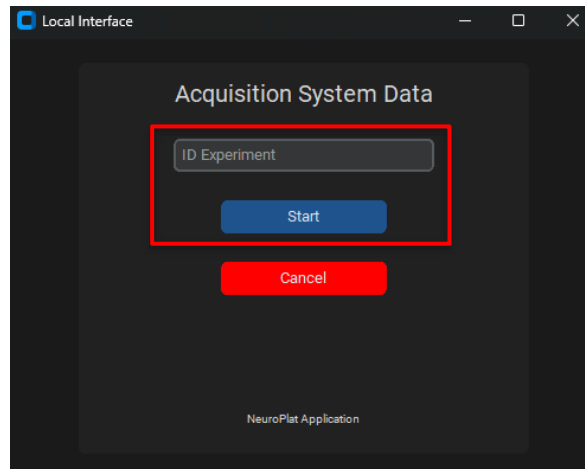


Figura 88. Interfaz de adquisición de señales con especificación de escritura del ID del experimento

**Paso 4:** Hacer Click en el botón “*Start*”

**Paso 5:** Verificar estados de acuerdo con el ID del experimento ingresado

En caso de que el experimento sea válido y la hora establecida no se haya cumplido, el sistema realizara el precargue de las ventanas para visualización de señales y generara el siguiente mensaje informativo en color verde.

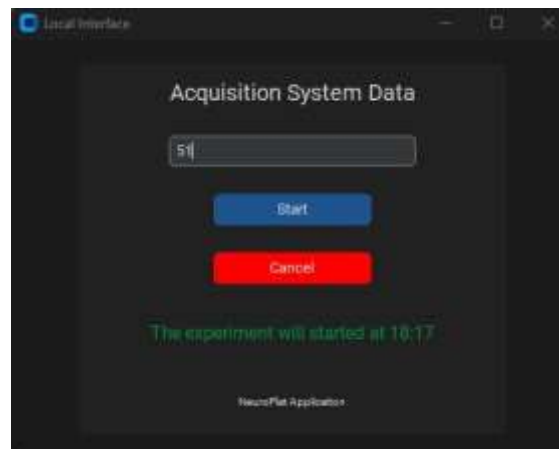


Figura 89. Mensaje de experimento encontrado y fecha de inicio de pruebas

Por el contrario, en caso de que la hora actual del ordenador exceda la hora establecida del experimento se solicitara al usuario la generación de un nuevo ID como se puede presentar a continuación.

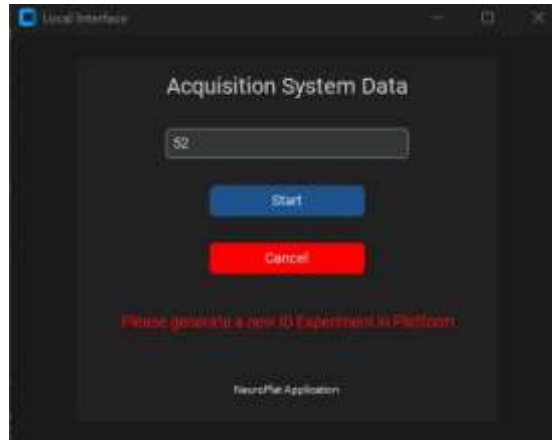


Figura 90. Mensaje informativo al usuario de expiración del ID del experimento

No se debe realizar ninguna acción adicional para transmisión de información el sistema generar automáticamente el almacenamiento y envío de datos a la plataforma. Para finalizar el proceso de adquisición, cierre la ventana o haga Click en el botón “*Cancel*”.

#### **ANEXO F. Repositorio de almacenamiento de código**

El código de la plataforma web (FrontEnd y BackEnd) y la interfaz de adquisición de señales es almacenado en la plataforma de control de versiones GitHub, la cual es posible acceder bajo la siguiente URL:

**URL Código:** <https://github.com/jcoronadop/NeuroPlat>

Es de recalcar que repositorio es de acceso privado, debido a que el código gestiona y controla información de usuarios, con sus respectivos procesos de autenticación. Dichos accesos serán controlados de acuerdo con las necesidades de implementación del proyecto y requerimientos técnicos establecidos que permitan cumplir con la normatividad legal vigente.

#### **ANEXO G. Documentación para conexión y uso API Telegram**

La documentación asociada con los procesos de conexión para envío y recepción de información por medio de chat de Telegram específicamente *bots* es obtenida de la página

oficial de la aplicación [34], allí se especifican las funciones y el procedimiento de conexión hacia la API.

Para generar la comunicación con el bot desde cualquier sistema o proceso se debe enviar en por medio de una solicitud API REST indicador el API Token correspondiente al *bot* creado como se especifica a continuación:

*https://api.telegram.org/bot{API\_Token}/getMe*

Los métodos dependiendo de las necesidades debe ser tipo GET para adquisición de datos y POST para envío de información.

Para la adquisición del ID y el nombre del propietario del bot se determinados por medio de la función “*User*”, necesarios para probar la comunicación.

Utilizando la función asociada “*message*” es posible enviar transmitir información tales como datos, imágenes, archivos y demás[34]

#### ANEXO H. Plan de pruebas del sistema.

Tabla 16. Plan de pruebas

Requerimiento Asociado	Objetivo de Prueba	Métrica	Procedimiento Ejecutado	Resultado
<b>RF01</b>	Visualización de señales EEG en la plataforma web	Las señales EEG deben mostrarse de forma gráfica y configurable para cada usuario	1. Ingresar al sistema como usuario registrado en la plataforma web desde la vista de login.	100% de las señales EEG se muestran correctamente y son visualizadas por canal
			2. Navegar a la sección de visualización de señales EEG.	
			3. Seleccionar una señal EEG de prueba para visualizar.	
			4. Verificar que la señal EEG se muestre de manera gráfica y configurable para el usuario.	

			5. Asegurarse de que sea posible personalizar la configuración de visualización según las preferencias del usuario.	
<b>RF02</b>	Acceso desde Navegadores y/o dispositivos móviles	La plataforma web debe ser accesible y funcional desde dispositivos móviles con acceso a internet	1. Acceder a la plataforma web desde un dispositivo móvil con acceso a internet. Desde la vista Login: Ingresar al sistema desde la vista de login en un dispositivo móvil. Desde la vista Dashboard: Navegar por el panel de inicio desde el dispositivo móvil.	100% de la plataforma es accesible por los usuarios y funcional en la comunicación dispositivos móviles
			2. Verificar que la plataforma web sea accesible desde el dispositivo móvil. Desde la Vista Dashboard: Comprobar la accesibilidad y el rendimiento de la plataforma en el dispositivo móvil.	
			3. Comprobar que todas las funciones principales estén disponibles y funcionales en el dispositivo móvil.	
<b>RF03</b>	Almacenamiento seguro del código fuente	El código fuente debe estar almacenado de manera segura en el repositorio en la nube	1. Verificar que el código fuente esté almacenado en el repositorio de código cerrado con control de versiones.	100% del código fuente está almacenado en GitHub como sistema de control de versiones y el acceso es restringido
			2. Comprobar que el acceso al repositorio esté restringido y solo las personas autorizadas puedan acceder a él.	
<b>RF04</b>	Registro y almacenamiento de mensajes	El sistema debe registrar y almacenar los mensajes cargados por los usuarios	1. Iniciar sesión como usuario registrado en la plataforma web. Desde la vista Login: Ingresar al sistema con credenciales válidas.	100% de los mensajes se registran y almacenan adecuadamente en la base de datos
			2. Cargar un mensaje en el sistema junto con la información del destinatario. Desde la vista de <i>Communication</i> validar que permita la establecer comunicación con el <i>bot</i> y permita realizar el envío de mensajes.	

			3. Verificar que el mensaje se registre y almacene correctamente en el sistema. Desde la vista <i>Communication</i> realizar la consulta de los mensajes almacenados en la base de datos de manera adecuada.	
<b>RF05</b>	Registro completo de experimentos	El sistema debe registrar todos los detalles de cada experimento, incluyendo información del paciente	1. Realizar un experimento utilizando señales EEG de prueba. Desde la vista de Experiments: Generar un nuevo experimento e ingresar información del paciente. Asegurarse de que todos los detalles del experimento, incluyendo la información del paciente, se registren de manera completa y precisa en el sistema.	100% de todos los detalles de los experimentos como precisión, calificación y secuencia se registran y almacenan correctamente
			2. Verificar que los datos del experimento se almacenen correctamente en la base de datos, incluyendo la hora de inicio y detalles del paciente.	
			3. Verificar que desde la vista de la Matrix los datos de las señales EEG adquiridas durante el experimento se almacenen de manera precisa y estén relacionados con el experimento.	
<b>RF06</b>	Gestión de usuarios registrados	La plataforma web debe permitir la gestión segura de usuarios registrados	1. Iniciar sesión como administrador de usuarios en la plataforma web desde la vista de login.	100% de las acciones de gestión se realizan de manera segura y la información se mantiene correctamente en la plataforma con acceso únicamente a super usuario
			2. Realizar acciones de gestión de usuarios, como agregar, modificar o eliminar usuarios registrados esta vista proporciona herramientas para administrar usuarios.	
			3. comprobar que las acciones de gestión se realicen de manera segura y que la información de los usuarios se mantenga correctamente y que la información de los usuarios se actualice correctamente en la base de datos.	
<b>RF07</b>	Multiplataforma	El sistema debe ser funcional en distintos sistemas operativos	1. Acceder a la plataforma web desde distintos sistemas operativos, como Windows, macOS, Linux, iOS y Android.	90% de los ingresos en sistemas operativos probados funciona de forma adecuada
			2. Acceso desde diferentes dispositivos: Realizar el acceso desde dispositivos que utilicen diferentes sistemas operativos.	



			3. Verificar que la plataforma web sea funcional en al menos el 90% de los sistemas operativos probados.	
			4. Comprobación de funcionalidad: Asegurarse de que la plataforma funcione adecuadamente en la mayoría de los sistemas operativos probados.	
<b>RF09</b>	Configuración de parámetros	Los usuarios deben poder configurar parámetros del sistema según sus necesidades	1. Ingresar al sistema como usuario registrado en la plataforma web desde la vista de login.	100% de los usuarios pueden configurar parámetros y los cambios se reflejan adecuadamente en cada una de las secciones del sistema
			2. Acceder a la sección de configuración de parámetros desde la vista de <i>Settings</i> seleccionar la sección de configuración de parámetros y modificar los parámetros del sistema según las necesidades del usuario.	
			3. Verificar que los cambios en los parámetros se reflejen correctamente en el sistema y afecten el funcionamiento según lo configurado desde la Vista Settings Comprobar que los cambios realizados en la configuración se apliquen de manera efectiva y se reflejen en el funcionamiento del sistema.	
<b>RF10</b>	Comunicación con tarjeta por integración	El sistema debe establecer una comunicación efectiva con la tarjeta por integración para adquirir señales EEG	1. Conectar la tarjeta por integración al sistema. - Realizar la conexión física de la tarjeta de adquisición de señales EEG. - Iniciar el sistema y establecer la comunicación con la tarjeta por integración para adquirir señales EEG.	100% de la comunicación es efectiva y las señales se adquieren y procesan correctamente entre todos los sistemas
			2. Verificar que la comunicación con la tarjeta por integración sea efectiva y que se adquieran las señales EEG correctamente.	
			3. Validar que el sistema pueda procesar y visualizar las señales EEG adquiridas de manera adecuada.	
<b>RF11</b>	Configuración del bot de comunicación	Validar que los cambios en la configuración del bot se guarden correctamente en el sistema y que el bot de comunicación utilice la información configurada para la transmisión de mensajes a través de la aplicación Telegram.	1. Ingresar al sistema como usuario registrado en la plataforma web desde la vista de "Login," introducir las credenciales de usuario para iniciar sesión. Acceder a la vista principal de la plataforma.	100% de los usuarios pueden configurar el bot individualmente y los cambios se guardan adecuadamente para conexión posterior

			2. Desde la seccion de "Comunicación" (Communication) en el menú principal el cual contiene las opciones de configuración del bot de comunicación.	
			3. En la sección de "Comunicación" (Communication), editar la información del bot, que puede incluir el nombre, dirección de contacto y otros detalles según las preferencias del usuario y guardar los cambios en la configuración del bot.	
			4. Validar de que los cambios realizados en la configuración del bot se guarden correctamente en el sistema.	
			5. Desde la vista de "Comunicación" (Communication), utilizar el bot configurado para enviar mensajes a través de la aplicación Telegram. - Verificar que los mensajes sean enviados correctamente.	
			6. Confirmar que los mensajes enviados desde la plataforma web a través del bot sean recibidos correctamente en la aplicación Telegram y que la información contenida en los mensajes sea precisa.	
<b>RNF01</b>	Seguridad de autenticación	Confirmar que el sistema rechaza el acceso con credenciales incorrectas y autentica al usuario de manera segura utilizando cifrado y tokens de acceso con credenciales válidas.	<p>1. Intentar acceder al sistema ingresando un nombre de usuario y una contraseña incorrectos. - Desde la vista de "Login," ingresar un nombre de usuario y una contraseña incorrectos y tratar de acceder al sistema. Verificar que el sistema rechace el acceso y muestre un mensaje de error adecuado.</p> <p>2. Confirmar que el sistema muestra un mensaje de error indicando que las credenciales son incorrectas y que el acceso está denegado. Intentar acceder al sistema con credenciales válidas.</p> <p>3. Desde la vista de "Login," ingresar un nombre de usuario y una contraseña válidos y tratar de acceder al sistema. Comprobar que el sistema autentique al usuario de manera segura utilizando cifrado y tokens de acceso</p>	100% del sistema autentica a los usuarios de manera segura y restringe el acceso no autorizado

			<p>4. Verificar que el sistema autentica al usuario de manera segura, utilizando cifrado y tokens de acceso para garantizar la seguridad de la autenticación.</p> <p>Asegurarse de que el acceso al sistema esté restringido a usuarios autorizados.</p> <p>Confirmar que solo los usuarios autorizados pueden acceder al sistema.</p>	
<b>RNF02</b>	Verificar que el sistema esté configurado para utilizar una única versión de funcionamiento y que no existan duplicaciones o instancias no autorizadas.	Confirmar que el sistema está configurado para utilizar una única versión de funcionamiento y que no hay duplicaciones de componentes ni instancias no autorizadas.	<p>1. Verificar que el sistema esté configurado para utilizar una única versión de funcionamiento.</p> <p>- Asegurarse de que el sistema esté configurado para utilizar una única versión de funcionamiento sin duplicaciones.</p>	100% del sistema utiliza una única versión de funcionamiento y no hay duplicaciones no autorizadas para uso o despliegue
			<p>2. Realizar un escaneo exhaustivo del sistema para identificar duplicaciones o instancias no autorizadas.</p> <p>- Escanear el sistema en busca de duplicaciones de componentes o instancias no autorizadas que puedan estar en funcionamiento.</p>	
			<p>3. Asegurarse de que no existan múltiples versiones en funcionamiento y que no haya duplicaciones de componentes.</p> <p>- Confirmar que no hay múltiples versiones en funcionamiento y que no existen duplicaciones de componentes en los sistemas de componentes</p>	
<b>RNF03</b>	Verificar el cumplimiento de la normatividad requerida para la tarjeta de integración utilizada en la medición no intrusiva de señales EEG.	Confirmar que la tarjeta de integración cumple con la normatividad requerida para medición no intrusiva en pacientes y que posee las certificaciones necesarias.	<p>1. Revisar la documentación de la tarjeta de integración para asegurarse de que cumple con la normatividad requerida para medición no intrusiva en pacientes.</p> <p>- Examinar la documentación proporcionada para la tarjeta de integración en busca de información sobre su cumplimiento con la normatividad requerida.</p>	100% de la tarjeta cumple con la normatividad y tiene las certificaciones necesarias
			<p>2. Verificar que la tarjeta de integración tenga las certificaciones y cumplimientos necesarios para su uso en aplicaciones médicas o de investigación.</p>	
			<p>3. Confirmar que la tarjeta de integración posee las certificaciones necesarias y cumple con los requisitos para su uso en aplicaciones médicas o de investigación.</p>	

<b>RNF04</b>	Verificar que el sistema cuente con mensajes informativos para guiar a los usuarios en su interacción con la plataforma.	Validar que el sistema muestre mensajes informativos relevantes, claros y concisos para guiar a los usuarios en su interacción con la plataforma.	1. Acceder al sistema como nuevo usuario o bajo condiciones que requieran mensajes informativos. - Iniciar sesión como nuevo usuario o simular condiciones que requieran mensajes informativos.	100% del sistema muestra mensajes informativos relevantes y claros en el momento adecuado para cada configuración o guardado
			2. Verificar que el sistema muestre mensajes informativos relevantes para guiar a los usuarios en su interacción con la plataforma. - Confirmar que el sistema muestra mensajes informativos que son relevantes para la situación y la interacción del usuario.	
			3. Comprobar que los mensajes informativos sean claros, concisos y proporcionen la información necesaria. - Validar la claridad y concisión de los mensajes informativos y asegurarse de que proporcionen la información necesaria para que los usuarios comprendan y utilicen el sistema de manera efectiva	
			4. Asegurarse de que los mensajes informativos se presenten en el momento adecuado y de acuerdo con el flujo de interacción del usuario. - Confirmar que los mensajes informativos se presentan en el momento oportuno y de acuerdo con el flujo de interacción del usuario.	
<b>RC02</b>	Validar la precisión del procesamiento de señales EEG de acuerdo con el paradigma P300 mediante pruebas con usuarios reales y comparación de datos.	Verificar la precisión del procesamiento, es decir, el porcentaje de señales EEG correctamente clasificadas de acuerdo con el paradigma P300, y confirmar que sea igual o mayor al 60%.	1. Identificar y seleccionar un grupo de usuarios voluntarios dispuestos a participar en las pruebas de procesamiento de señales EEG.	70,5% de precisión en la clasificación de acuerdo con las pruebas realizadas
			2. Iniciar el sistema y configurar los parámetros de procesamiento de acuerdo con las especificaciones del paradigma P300. Realizar pruebas con los usuarios, registrando sus señales EEG mientras se someten a diferentes estímulos y situaciones.	

			3. Realizar pruebas con cada usuario, registrando sus señales EEG mientras se someten a diferentes estímulos y situaciones de acuerdo con el paradigma P300.	
			4. Procesar y clasificar las señales EEG registradas para cada usuario. - Utilizar el sistema para procesar y clasificar las señales EEG registradas para cada usuario en las pruebas. Registrar los resultados de clasificación de las señales EEG para cada usuario.	
			5. Registrar los resultados de la clasificación de las señales EEG para cada usuario en las pruebas.	
			6. Calcular la precisión del procesamiento por separado para cada usuario, es decir, el porcentaje de señales EEG correctamente clasificadas de acuerdo con el paradigma P300.	
			7. Realizar una comparación de los datos de precisión entre los diferentes usuarios. - Comparar los datos de precisión obtenidos de las pruebas realizadas con distintos usuarios para validar la consistencia y rendimiento del sistema.	
			8. Confirmar que la precisión promedio del procesamiento sea igual o mayor al 60% a través de las pruebas con usuarios reales. - Calcular la precisión promedio del procesamiento considerando los resultados de todas las pruebas con usuarios voluntarios y asegurarse de que sea igual o mayor al 60%	
<b>RC03</b>	Garantía de envío y transmisión de datos	Validar que los mensajes sean enviados desde la plataforma web hacia la aplicación Telegram y que sean recibidos correctamente en la aplicación Telegram. Además, asegurarse de que no haya pérdida de datos durante el proceso de envío y transmisión de mensajes.	1. Ingresar al sistema como usuario registrado en la plataforma web desde la vista de "Login," introducir las credenciales de usuario para iniciar sesión.	100% de los mensajes se envían y reciben correctamente sin pérdida de datos
			2. Acceder a la sección de comunicación desde la plataforma web.	
			3. Ingresar un mensaje en la plataforma web para ser enviado a través de la aplicación Telegram.	

			4. Comprobar que el mensaje sea enviado desde la plataforma web hacia la aplicación Telegram.	
			5.Verificar que el mensaje enviado desde la plataforma web sea transmitido con éxito a la aplicación Telegram.	
			6. Acceder a la aplicación Telegram y asegurarse de que el mensaje enviado desde la plataforma web sea recibido de manera precisa y completa.	
			7.Realizar varias rondas de envío y recepción de mensajes para confirmar que no se haya pérdida de datos en ningún punto del proceso.	