

## Trabajo Fin de Máster

Análisis y tratamiento en tiempo real de la señal  
de un electroencefalograma con fines  
educativos

Streaming analysis and processing of an  
encephalogram signal for educational purposes

Autora

Claudia Gonzalo Gimeno

Directores

Jorge Luis Falcó Boudet

Isidro Urriza Parroqué

ESUELA DE INGENIERÍA Y ARQUITECTURA

2020

## Resumen

La detección y diagnóstico de trastornos neurológicos, como el déficit de atención con hiperactividad, es una tarea costosa que requiere mucho tiempo. La electroencefalografía es un método no invasivo de medida de la actividad cerebral en continua investigación, que permite detectar este trastorno y evaluar el estado del paciente.

La inclusión de esta herramienta en el aula de educación especial facilita el seguimiento del alumno, ya que el aprendizaje es un proceso continuo y dinámico que ha de ir adaptándose a las necesidades de este. Para su aplicación surgen dos necesidades: una, que el proceso de “preparación del alumno y los cables de las conexiones de los electrodos” sea menos costoso en tiempo y molestia al alumno que los sistemas BCI convencionales; y la segunda, que podamos obtener la información de forma dinámica para modular el esfuerzo de comprensión o la tarea docente que se plantea en función del estado del alumno.

En el presente proyecto se emplea el electroencefalograma Bitmed eXea Ultra. La dificultad del uso de este aparato tal y como se distribuye comercialmente es la imposibilidad de acceder y procesar los datos de forma dinámica, pues únicamente permite su almacenamiento y posterior procesado. Por ello la acción que se plantea en este proyecto es la implementación de la comunicación directa mediante comandos con el ordenador. Una vez establecida, se detectan los parámetros más importantes que aportan información sobre el estado del alumno y se realiza su correspondiente representación gráfica.

Finalmente se integra todo lo anterior en una aplicación, con la característica diferenciadora de emplear únicamente dos electrodos para realizar los registros del electroencefalograma. Esto conlleva la reducción de los problemas mencionados, a la vez que permite al experto hacer un análisis más detallado. De esta forma se asegura el proceso de enseñanza, facilitando la labor del experto en cuestión.

Esta aplicación consta de dos modos. El primero de ellos, permite recoger los datos del electroencefalograma en tiempo real y realizar gráficos que se actualicen automáticamente conforme se registran. El segundo modo permite cargar archivos de registros anteriores para representar de forma paralela diferentes gráficos y, mediante barras deslizadoras, poder analizar más detenidamente un evento o característica de los datos.

A lo largo de esta memoria se explican las diferentes fases que han compuesto este proyecto, las decisiones tomadas y los resultados obtenidos. Finalmente, se incluye un apartado de conclusiones y líneas futuras de trabajo.

# Índice

1	Introducción .....	1
1.1	Contexto, objetivos y planteamiento del problema .....	1
1.1.1	Metodología .....	2
1.2	Herramientas .....	2
1.2.1.1	Herramientas de desarrollo.....	3
1.2.1.2	Lenguajes de programación.....	3
2	Marco de la acción de investigación: selección de la tecnología .....	4
2.1	Métodos de medida de la actividad cerebral .....	4
2.1.1	Infrarrojo de espectro cercano.....	4
2.1.2	Magnetoencefalografía.....	5
2.1.3	Tomografía por emisión de positrones.....	5
2.1.4	Electroencefalografía .....	6
2.2	Ámbito educativo .....	7
2.2.1	Electroencefalografía para la educación.....	8
2.2.2	Aplicación del proyecto .....	10
3	EEG: Bitmed eXea Ultra.....	11
4	Registro del EEG.....	14
4.1	Comandos.....	15
4.1.1	Empezar una prueba en tiempo real .....	16
4.2	Proceso de recogida de datos .....	19
5	Procesado de los datos EEG y su representación gráfica .....	21
5.1	Visualización 2D.....	21
5.2	FFT.....	22
5.2.1	Diagrama de sectores y gráfico de barras.....	23
5.3	Cartografía cerebral.....	24
6	Aplicación .....	25
6.1	Análisis.....	25
6.1.1	Modos.....	25
6.1.2	Requisitos funcionales.....	25
6.1.3	Requisitos no funcionales.....	26
6.2	Diseño .....	26
6.2.1	Arquitectura.....	26
6.2.2	Diagramas de flujo .....	28
6.2.3	Bocetos .....	30
6.3	Implementación.....	32
6.3.1	Lanzamiento.....	32
6.3.2	Guardado de datos .....	33

6.3.3	Hilos .....	33
6.3.4	Representación gráfica .....	34
6.3.5	Atributos de calidad .....	36
7	Pruebas .....	37
7.1	Prueba final .....	38
8	Repositorio .....	39
9	Conclusiones .....	40
9.1	Futuras líneas de investigación .....	40
10	Glosario .....	42
11	Bibliografía .....	42
12	Índice de tablas.....	47
13	Índice de ilustraciones.....	47
14	Índice de ecuaciones.....	48
15	Anexos.....	48

# 1 Introducción

En este apartado se describe el punto de partida del proyecto, con sus objetivos y la forma de abordarlos, se explica la metodología empleada, así como las herramientas y lenguajes utilizados. Se incluye un cronograma que temporiza el proyecto.

## 1.1 Contexto, objetivos y planteamiento del problema

Este TFM se enmarca en una acción de investigación y desarrollo que busca dotar al docente de educación especial de una medida instantánea del funcionamiento mental del alumno.

En principio, el proyecto se centra en parámetros de hiperactivación o hipoactivación mental. La hiperactivación reduce la capacidad de eficacia y aprendizaje por estrés, y la hipoactivación se relaciona con la bajada de motivación del alumno y por tanto de atención reducida sobre la tarea que se le propone. La topología de las zonas del cerebro que se activan, hiperactivan o hipoactivan, pueden ser relevantes dada la relación de activación de cada zona con una significancia funcional (procesamiento paralelo, procesamiento serie, estrategias, empatía, etcétera).

El objetivo primario de facilitar esta medida del funcionamiento mental del alumno es permitir un ajuste de la dificultad y/o del atractivo de la tarea a realizar para mantenerle en una zona de funcionamiento óptima.

- Se plantea el desarrollo de un instrumento que permita utilizar un feedback del funcionamiento mental de un alumno de educación especial en tiempo real para el ajuste dinámico de la dificultad o grado de abstracción, demanda o recompensa que se adecúe a su estado.
- Para ello primero se realiza un estudio de distintas tecnologías de captación de información del funcionamiento cerebral.
- Se observa la versatilidad e idoneidad del EEG, en especial con las aportaciones de procesamiento matemático que infieren el grado de hiperactividad con las señales de tan sólo dos electrodos en el lóbulo frontal del alumno.

Por otra parte, y fruto de proyectos anteriores, se dispone de un aparato de EEG conectable al ordenador vía serie, y su configuración mediante comandos (Exea Ultra de Bitmed), y de información técnica de sus protocolos de comunicación con el ordenador, aunque con limitaciones importantes atentos a los objetivos perseguidos:

- En la aplicación que acompaña al EEG, el acceso a los datos instantáneos se ofrece tan sólo a nivel gráfico de amplitud de cada canal frente al tiempo. El acceso a los datos para permitir otro tipo de procesados se realiza tras la grabación de la sesión completa en el ordenador, lo que imposibilita el funcionamiento dinámico requerido.
- La casa fabricante cambió su estrategia de software hace ya años y no ofrece disponibilidad de un módulo de comunicación con el ordenador, ni información adicional necesaria.
- La información que se solicitó en su día al fabricante permite una conexión con el ordenador a bajo nivel, aunque las pruebas realizadas dejan incertidumbres en algunos puntos que suponen un reto para la consecución de la comunicación serie con el ordenador, siendo por tanto este el primer objetivo a conseguir.
- La máquina se comercializó en el año 2010, siendo compatible, para su empleabilidad, con el sistema operativo Windows 7. Una vez se desarrollaron los siguientes sistemas operativos se encuentran incompatibilidades con los drivers de los programas que permiten la comunicación entre el ordenador y la máquina.

Por tanto, se plantean los siguientes objetivos tecnológicos:

- Conexión del EEG al PC: pruebas de comunicación para completar el protocolo que lleva el EEG para su control y acceso desde el ordenador.
- Grabación directa de los datos en un fichero, para acceso inmediato por parte de programas de procesamiento.
- Disponibilidad de una primera versión de datos procesados en una interfaz sencilla de usuario, como muestra de las posibilidades del proyecto acometido.

### 1.1.1 Metodología

De esta manera, para el cumplimiento de los objetivos planteados, se enumeran a continuación las diferentes fases que conforman el proyecto:

- Estudio sobre los diferentes métodos de medida de la actividad cerebral, centrándose en el ámbito de la educación.
- Estudio sobre la herramienta EEG.
- Estudio sobre el protocolo de comunicación del EEG con el ordenador.
- Estudio sobre el formato de los datos ofrecidos por el EEG.
- Estudio sobre la información útil ofrecida por los parámetros del EEG en el ámbito educativo.
- Estudio de los diferentes tratamientos y formas de representar los datos obtenidos.
- Estudio e implementación de una solución software que permita representar los datos en sus diferentes formas, con el fin de enmarcarlos en el entorno especial educativo.

Paralelamente a todo esto, se redacta la memoria para dejar constancia de los resultados obtenidos en el proyecto. A continuación, en la Tabla 1 se ofrece un cronograma que temporiza las tareas mencionadas anteriormente en su correspondiente orden.

	<b>Meses (Años 2019-2020)</b>											
<b>Tareas del proyecto</b>	Dic	En	Feb	Mar	Abr	May	Jun	Jul	Ag	Sep	Oct	Nov
Estudio sobre los métodos de medida												
Estudio sobre la herramienta EEG												
Estudio protocolo de comunicación												
Estudio formato de los datos												
Estudio parámetros del EEG												
Representación de los datos												
Implementación solución software												
Documentación de la memoria												

Tabla 1: Cronograma

## 1.2 Herramientas

Para desarrollar el proyecto se han empleado distintas herramientas de desarrollo con sus correspondientes lenguajes de programación, las cuales se detallan a continuación.

### 1.2.1.1 Herramientas de desarrollo

- **ConnTest:** Es una aplicación proporcionada por el fabricante que permite intercambiar comandos básicos con el EEG. Se emplea para, mediante un osciloscopio, observar las tramas que entran y salen del transmisor-receptor asíncrono universal (UART) al inicio del proyecto.
- **SerialPlot:** Es una aplicación para representar datos desde el puerto serie en tiempo real [1]. Se selecciona el puerto y se configura, posteriormente se puede enviar-recibir datos. Se emplea para determinar el baud rate del EEG.
- **Anaconda:** Es un kit de herramientas, de ciencia de datos, de código abierto. Desarrollado para practicantes en solitario, posee un conjunto de herramientas que lo equipa para trabajar con miles de paquetes y bibliotecas, la interfaz de usuario facilita el aprendizaje. Anaconda Navigator es una GUI de escritorio que facilita el lanzamiento de aplicaciones y la administración de paquetes y entornos sin usar comandos de línea de comandos [2]. Para este proyecto se ha empleado Visual Studio Code, un editor de código fuente incluido en Anaconda Navigator que incluye soporte para depuración, control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código [3]. Todos los lenguajes mencionados en el apartado siguiente son soportados por esta herramienta.
- **Electroencefalograma Bitmed eXea Ultra:** Es el hardware del proyecto, se explica en el apartado EEG: Bitmed eXea Ultra.

### 1.2.1.2 Lenguajes de programación

- **Python:** Es un lenguaje de programación, independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa. Es un lenguaje interpretado, lo que ofrece ventajas, como la rapidez de desarrollo, e inconvenientes, como una menor velocidad [4]. Se ha empleado para programar los diferentes ‘scripts’ de la aplicación, los relativos a los gráficos y el principal. En este proyecto se utiliza el framework Flask para desarrollar la aplicación web tal y como se detalla en el *Anexo 1*, donde encontramos también una descripción de las bibliotecas utilizadas. La versión empleada es: Python 3.8.5.
- **HTML:** HTML es el lenguaje que se emplea para el desarrollo de páginas web. Está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla. Dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas, tablas, etcétera [5]. Se ha empleado en la aplicación para definir la estructura de las diferentes vistas que puede manejar el usuario.
- **CSS:** Es lo que se denomina ‘lenguaje de hojas de estilo en cascada’ y se emplea para estilizar elementos escritos en un lenguaje de marcado como HTML. CSS separa el contenido de la representación visual del sitio [6]. Se ha utilizado para dotar de estilo a las vistas de la aplicación, añadiendo colores, formas, etcétera. Se ha incorporado el framework CSS Bootstrap 4 por sus plantillas de diseño, botones, selección entre diferentes opciones...
- **JavaScript:** Es un lenguaje gracias al cual se desarrolla lo relacionado con la funcionalidad en sitios y aplicaciones web modernas. En el proyecto se ha empleado para que el usuario pueda interactuar con la aplicación y así capturar los datos que introduce el usuario en esta.

## 2 Marco de la acción de investigación: selección de la tecnología

### 2.1 Métodos de medida de la actividad cerebral

Si se realiza un análisis de los datos estadísticos sobre los trastornos neurológicos ofrecidos por diferentes países, podemos determinar que hay millones de personas en todo el mundo que sufren alguna enfermedad neurológica que puede cambiar su vida de una manera muy negativa. Por ejemplo, la prevalencia del trastorno de déficit de atención con hiperactividad en niños de 18 años y menores tiene una estimación global combinada de 7.2% [7]. Otro ejemplo lo encontramos en que alrededor del 1% de la población mundial sufre epilepsia y la aparición de ataques repentinos e impredecibles puede convertirse en una causa de lesión o discapacidad [8].

El diagnóstico actual de trastornos neurológicos es una tarea costosa, que requiere mucho tiempo. El cerebro posee numerosas señales biológicas que proporcionan valiosa información sobre el estado del paciente, y es por ello que, tanto los diferentes métodos de medida de la actividad cerebral, como las herramientas que facilitan su diagnóstico están, en continua evolución.

Actualmente los métodos más investigados son los no invasivos, ya que no implican la realización de un procedimiento quirúrgico. Sobre ellos se están desarrollando numerosas aplicaciones para poder detectar y tratar los diferentes trastornos. Algunos de estos métodos son: infrarrojo por espectro cercano, magnetoencefalografía, tomografía por emisión de positrones o electroencefalografía.

#### 2.1.1 Infrarrojo de espectro cercano

Se utiliza la espectroscopía de infrarrojo cercano (NIRS) para medir la función cerebral. Es un método que mide el cambio de concentración de oxi-hemoglobina mediante el uso de las características de la luz infrarroja cercana. Si esta se irradia sobre el cuero cabelludo, alcanza la corteza cerebral en las profundidades de 20 a 30 mm. La principal característica de la hemoglobina, para la luz infrarroja cercana, es una absorción diferente de la que es oxigenada respecto de la desoxigenada. Si se detecta la intensidad de la luz entre diferentes puntos y el punto de irradiación, es posible medir los cambios en la concentración de hemoglobina. Por lo tanto, con NIRS es posible medir la función cerebral, dado que las zonas más activas consumen más oxígeno y se diferencian las concentraciones de oxi-hemoglobina y carboxi-hemoglobina [9].

Mediante el uso de la colocación de electrodos, se determinan los puntos de medición. En la Ilustración 1 podemos observar un ejemplo de cómo posicionarlos, empleando el denominado sistema diez y veinte.

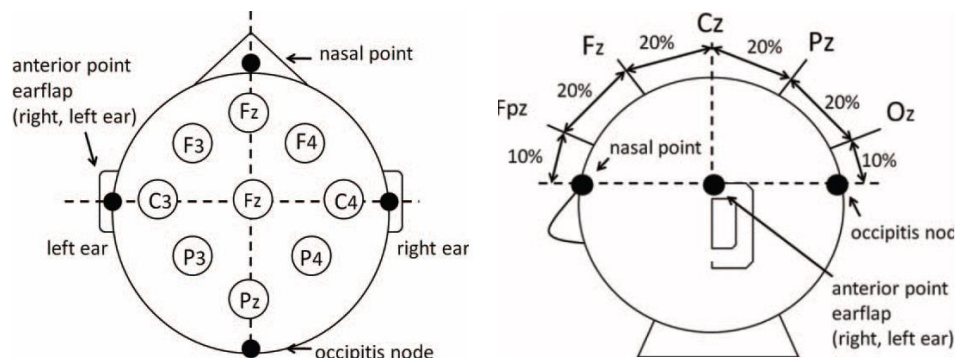


Ilustración 1: Sistema de colocación de electrodos diez y veinte. (a) Vista superior. (b) Vista lateral. [9].

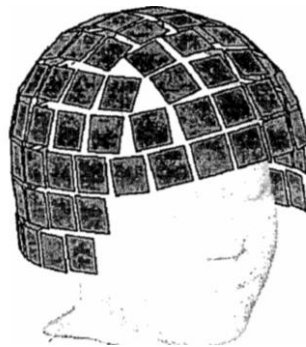


Este método también permite medir la oxigenación [10] y comprender los efectos de la hipertensión inducida por el embarazo sobre recién nacidos [11]. Otro de sus ámbitos de investigación es la nutrición [12], [13].

### 2.1.2 Magnetoencefalografía

La magnetoencefalografía (MEG) es una herramienta no invasiva capaz de investigar la dinámica de la actividad neuronal en el cerebro humano vivo, proporcionando una nueva fuente de información para analizar las actividades cerebrales. Este método mide los campos magnéticos producidos por las corrientes eléctricas del cerebro [8], [14].

En [14] la Ilustración 2 muestra un ejemplo de la matriz de sensores MEG diseñados para dicha investigación y su posicionamiento sobre el sujeto a estudiar. Las placas cuadradas representan los sensores empleados para medir el pertinente campo magnético.



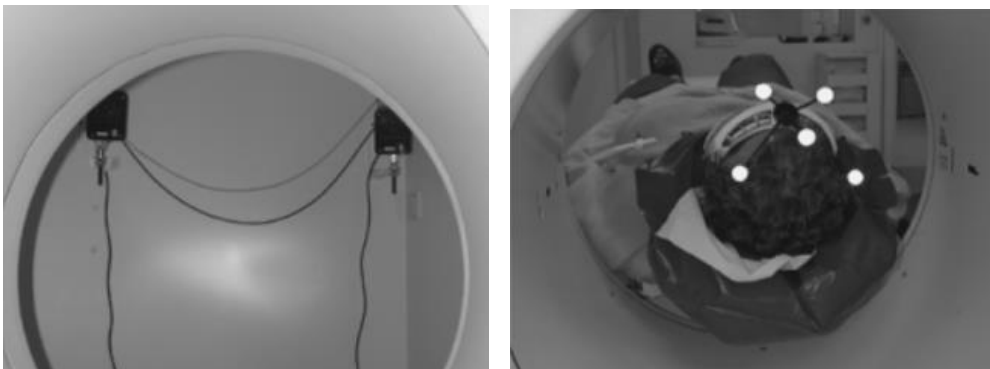
*Ilustración 2: Configuración del conjunto de sensores SQUID [14].*

Actualmente, MEG se está utilizando para el mapeo funcional no invasivo de pacientes epilépticos, para la evaluación prequirúrgica y la localización de descargas epilépticas. Hay que mencionar que los trabajos en este área son escasos, lo que puede deberse en parte al alto costo de la tecnología MEG y de ahí su disponibilidad limitada en todo el mundo [8], [14].

### 2.1.3 Tomografía por emisión de positrones

La tomografía por emisión de positrones (PET) se basa en captar la radiación emitida por una sustancia inyectada al paciente de carácter radiactivo (radiofármaco) con la finalidad de poder analizar su distribución y su metabolización, lo que permite el seguimiento de su actividad cerebral.

PET se emplea para estudiar diversas enfermedades cerebrales humanas, incluida la demencia de Alzheimer y su riesgo genético, prediciendo el inicio y controlando la progresión de la enfermedad. También se puede emplear para determinar el volumen de un tumor, por el hecho de que las células cancerosas tienen un valor diferente de absorción del isótopo radiactivo que las células normales [15], [16].



*Ilustración 3: Sistema de seguimiento del paciente en escáner PET. (a) Vista de frente. (b) Vista del paciente. [16]*

La corrección del movimiento del paciente durante las investigaciones de PET es un problema persistente. El movimiento del paciente puede alterar severamente la validez de las imágenes reconstruidas, el movimiento del cabezal durante los escaneos da lugar a una fuerte degradación de la imagen: en la Ilustración 3 se encuentra un ejemplo del sistema de seguimiento del paciente en el escáner. Esto hace que sea un método relativamente de larga duración, por ello, se están investigando diferentes métodos y algoritmos que corrijan los movimientos y ofrezcan una mayor precisión [17], [18], [19], [20].

#### 2.1.4 Electroencefalografía

La electroencefalografía lee la actividad eléctrica de las estructuras cerebrales sobre el cuero cabelludo. El electroencefalograma es el método preferido para medir la actividad cerebral de manera no invasiva en aplicaciones de la vida real. Las señales de EEG se registran en el cuero cabelludo utilizando múltiples electrodos y medios conductores en respuesta a estímulos específicos, como: señales auditivas o visuales inesperadas, estímulos visuales rítmicos o la imaginación de movimientos [21], [22], [23].

Normalmente se colocan 32 electrodos siguiendo el sistema de colocación de electrodos estándar internacional 10-20, tal y como muestra la Ilustración 4.

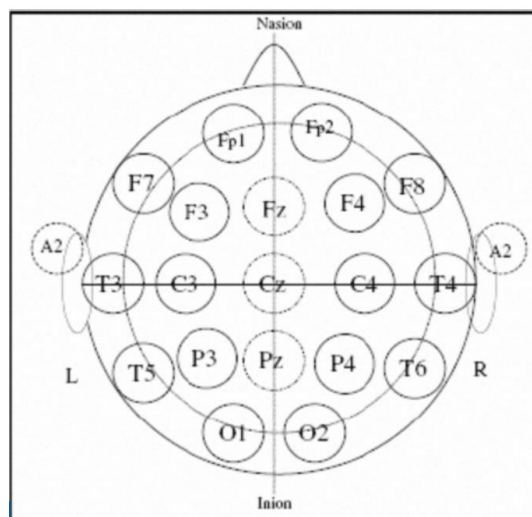


Ilustración 4: Sistema internacional 10-20 de colocación de electrodos en EEG. [24].

Cada registro se ve como un voltaje en función del tiempo y cada canal se corresponde con una diferencia de potencial eléctrico entre electrodos. Se miden los potenciales eléctricos generados por los potenciales postsinápticos excitadores e inhibitorios, desarrollados por los cuerpos celulares y las dendritas de las neuronas piramidales, podemos ver su estructura y generación de la señal en la Ilustración 5 [25].

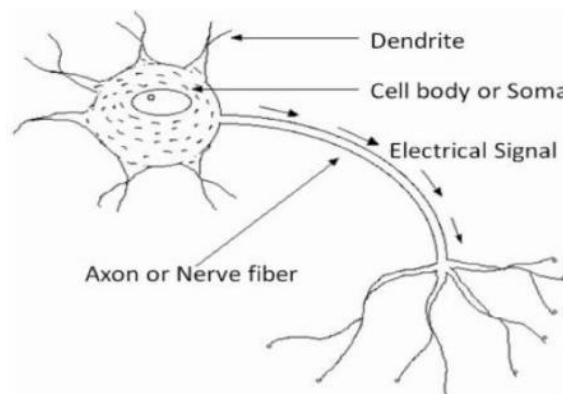


Ilustración 5: Estructura general de la neurona y generación de señal eléctrica debido a estimulación externa [25].

Las bandas de frecuencia de EEG están asociadas con diferentes características de actividad cerebral que cambian de acuerdo con diferentes situaciones. Las características se extraen y analizan para obtener una representación topológica del funcionamiento del sistema nervioso [22].

Los sensores húmedos, la transmisión por cable y la densidad de estos, son las principales barreras en la mayoría de los sistemas EEG. Para los electrodos húmedos, hay que insertar un gel entre los electrodos y el cráneo para reducir su impedancia, además de que, la preparación de la piel para aplicar este tipo de electrodos es de aproximadamente entre 30 - 45 minutos. Para superar estos problemas, se han desarrollado algunos tipos de novedosos sistemas EEG con electrodos secos y transmisión inalámbrica [21], [25]. Podemos observar un ejemplo en la Ilustración 6, donde se ha insertado un globo entre la tapa y el sensor para proporcionar la presión necesaria, de esta forma la distancia entre los electrodos secos y el cráneo se puede reducir.



Ilustración 6: Sistema de medición EEG. (a) Características del sistema, vista de frente. (b) Globo de presión, vista superior. [21].

Estudios de investigación desvelan que, el análisis de los datos obtenidos de EEG, permite reconocer emociones humanas, diagnosticar trastornos neurológicos, predecir el interés de una persona, discriminar pacientes con esquizofrenia o Alzheimer, reconocer ataques epilépticos... [8], [22], [24], [26], [27], [28], [29], [30], [31]. Por lo que es de gran interés su continua investigación.

Los métodos de medida mencionados anteriormente además de emplearse frente a un estímulo visual, sonoro, motor, etcétera, se combinan con diferentes modalidades de estimulación cerebral directa, como puede ser el caso de los ultrasonidos, la estimulación cerebral profunda o la estimulación magnética y eléctrica transcraneal [32], [33].

Además, para obtener un análisis más detallado, u obtener diferentes puntos de vista, se combinan varios de estos métodos, como es el caso de EEG y MEG. Se han propuesto numerosos procesos de localización de fuentes basados en estos métodos, ya sea para comprender mejor las funciones normales del cerebro o para localizar, de manera no invasiva, el origen de las señales anormales [34]. Durante la última década, esta combinación se ha convertido en una de las herramientas de diagnóstico que contribuye a una mejor selección de pacientes en el proceso hacia la cirugía de epilepsia [35].

## 2.2 Ámbito educativo

Si nos centramos en el ámbito de la educación, podemos destacar dos de los métodos ya comentados: infrarrojo de espectro cercano y electroencefalografía, siendo de gran ayuda para detectar y tratar los diferentes trastornos neurológicos asociados al lenguaje, ofreciendo nuevas herramientas de ayuda en este ámbito.

Gracias a ellos se presenta una nueva forma de aprender para el alumno que ofrece un mejor seguimiento de este para el docente. Ofrecen una nueva perspectiva del estado mental del alumno

permitiendo que, mediante ejercicios, pueda evolucionar de un estado a otro y así asegurar el mejor aprendizaje de conceptos y habilidades.

Las investigaciones sobre el primero de ellos, el infrarrojo cercano, se utilizan en el ámbito educativo más general. En la escuela, la gestión del aula es el factor más importante en el proceso educativo y el requisito básico de los docentes, los maestros enfrentan dificultades para hacer que todos los estudiantes mantengan la atención, porque la actividad mental y el problema personal de cada estudiante son diferentes. En [36] crean un sistema de observación del estrés para que los maestros monitoricen el nivel de estrés cerebral de los estudiantes mediante el uso del sensor NIRS HOT-1000 de espectroscopia de infrarrojo cercano (NIRS) de 2 canales, el cual podemos observar en Ilustración 7. Así pueden mejorar su eficiencia de enseñanza y adaptarla para cada estudiante. El resultado de dicha investigación describe que el promedio del valor de diferencia de la oxihemoglobina fue mayor cuando ocurrió un alto estrés.



*Ilustración 7: Sensor NIRS HOT-1000. [36].*

También se ha empleado para esclarecer la hipótesis "ventaja bilingüe". Esta hipótesis apunta a que los niños bilingües ofrecen un mejor desempeño en las tareas de la función ejecutiva en relación con los monolingües. En [37] los alumnos fueron conducidos a una tarea de atención no verbal mediante una técnica de espectroscopia de infrarrojo cercano. El estudio muestra que la experiencia bilingüe podría acelerar la ventaja cognitiva y modificar el cerebro funcional en la tarea de control atencional. Así podemos comprender cómo el aprendizaje de idiomas extranjeros impacta el desarrollo del cerebro de los niños.

### 2.2.1 Electroencefalografía para la educación

El desarrollo tecnológico de la electroencefalografía y sus investigaciones, se centran más en la educación especial. El hecho de que el aprendizaje sea un proceso continuo y dinámico significa que los programas educativos deben ajustarse con frecuencia en función de los resultados relacionados con su progreso. Las ondas cerebrales están altamente correlacionadas con el estado mental de uno, incluido el nivel de concentración, el estado emocional y el grado de relajación [38].

Como la señal del EEG captura la actividad eléctrica del cerebro, es una importante fuente de información para estudiar los trastornos neurológicos. En el marco educativo encontramos el reto de detectar y tratar los diferentes trastornos cognitivos, destacando la hiperactividad y el autismo. Se presenta como una importante herramienta para implementar nuevas terapias neuroprotectoras y novedosos métodos educativos [38], [24].

En la edad infantil, los trastornos cognitivos múltiples se basan en el síndrome de TDAH (trastorno por déficit de atención e hiperactividad). Muchos investigadores lo conectan con lesiones orgánicas cerebrales en las primeras etapas de la ontogenia. Este trastorno conduce a formas anormales de conexiones de los sistemas intracerebrales. Bajo este síndrome, aparecen trastornos del desarrollo de las funciones ejecutivas, como la atención, la memoria de trabajo, el habla, la praxis dinámica, la coordinación visual y auditiva, la planificación de la acción y la determinación ... [39].

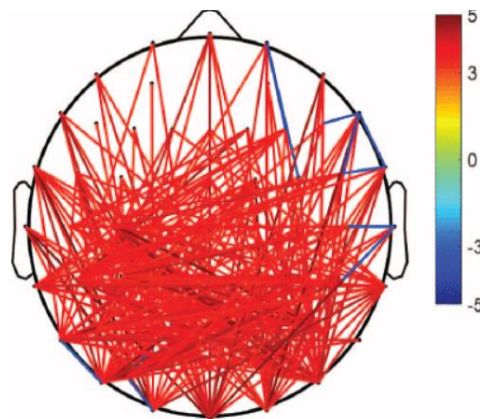
El autismo, nombrado técnicamente como Trastorno del espectro autista (TEA), es uno de los cinco trastornos del desarrollo generalizado que hace que una persona tenga graves dificultades en las áreas del desarrollo cognitivo y social. Pertenecer a una categoría de trastorno del desarrollo neurológico y trae discapacidades en la comunicación, el aprendizaje, el desarrollo y las habilidades sociales [40].

Las lesiones orgánicas del sistema nervioso central (SNC), como resultado de varias razones, son factores importantes en el desarrollo de déficits cognitivos a cualquier edad. El enfoque más común para el tratamiento de estas enfermedades es una terapia neuroprotectora que mejora la actividad del tejido nervioso y su recuperación estructural en respuesta a los patógenos emergentes. La farmacoterapia implica el uso de una gran cantidad de medicamentos que no solo interactúan entre sí y no son efectivos, sino que también interfieren en el curso natural de los procesos reguladores y adaptativos del SNC, actuando como un factor de estrés. En los últimos años surgen otros métodos no farmacológicos de efectos neuroprotectores, y una forma de verificar y monitorizar sus efectos es emplear el EEG como herramienta, por ejemplo, en [39] emplean la neuroelectroestimulación con la herramienta EEG.

Tras analizar los resultados que el EEG ofrece conforme se realiza la terapia, se pueden modificar, adaptar y verificar los parámetros y estímulos de esta. Estas terapias son usadas para influir en el deterioro cognitivo en niños y así curar trastornos depresivos y de ansiedad y trastornos infantiles, por ejemplo, trastorno por déficit de atención e hiperactividad, tartamudeo, enuresis, autismo... [39].

Numerosos estudios muestran los beneficios de emplear la herramienta EEG para el análisis, detección y/o tratamiento de los trastornos por déficit de atención o hiperactividad [27], [28], [41], [42], [43], [44], [45], [46], [47], [48], [49]. Además, con ella se mejora el conocimiento sobre estos trastornos y permite establecer una serie de pautas para su tratamiento.

El análisis de los datos obtenidos se puede realizar de forma visual gracias a los mapas topográficos que muestran las conexiones entre los diferentes electrodos. Un ejemplo lo encontramos en [28], en la Ilustración 8.



*Ilustración 8: Ejemplo mapa topológico.*

Los desarrollos recientes en la tecnología EEG, como los electrodos secos y la transmisión inalámbrica, han permitido aplicar estos dispositivos en el campo de la educación para monitorear el estado mental de los estudiantes que participan en el aprendizaje [38].

El EEG también se puede emplear en el aula normalizada, tanto en educación superior como en educación obligatoria. Un ejemplo en educación obligatoria lo encontramos en [50], donde se desarrolla un modelo educativo práctico en el que los estudiantes aprenden a diseñar y construir su propio sistema de interfaz cerebro-computadora (BCI) midiendo su propia actividad cerebral. La medición de las señales EEG le permite al usuario comunicarse con una computadora. Este documento también examina una implementación práctica del sistema que involucra una composición de música generada por EEG.

Por otro lado, un ejemplo en educación superior lo encontramos en [38], donde este estudio desarrolla un sistema de aprendizaje adaptativo basado en señales EEG. El aprendizaje adaptativo es un enfoque prometedor para la educación, en el que los métodos y materiales de instrucción se seleccionan de acuerdo con el desempeño de los estudiantes. El estudio se basa en monitorizar el proceso de los estudiantes mediante ondas cerebrales para permitir el ajuste dinámico de métodos y / o materiales de instrucción dentro de un entorno de aprendizaje electrónico que se adapta a las necesidades y fortalezas de los estudiantes para maximizar su eficiencia. Utilizamos un dispositivo EEG inalámbrico para medir las ondas cerebrales y luego analizar su respuesta espectral.

Hay que mencionar, que el éxito en la resolución y la aplicación de todos métodos nombrados anteriormente dependen de la interacción y colaboración de diferentes especialistas: ingenieros, médicos, fisiólogos, biólogos, educadores, etcétera, ya que además de la investigación hay que generar prototipos funcionales que faciliten la labor de aprendizaje del alumno.

### 2.2.2 Aplicación del proyecto

El presente TFM se centra en la educación especial, en los niños que sufren TDHA, concretamente en la electroencefalografía, ya que es el método más empleado en dicha educación. Se ha comentado anteriormente que el tiempo estimado de preparar a un niño para un registro de EEG es elevado, además colocar los electrodos con su correspondiente gel en una determinada posición se complica debido a esa falta de atención e hiperactividad. Hay que mencionar que los equipos inalámbricos y de simple uso mencionados son escasos, de precio elevado y la mayoría pertenecen al ámbito de la investigación, no comercializándose.

Se hace latente la necesidad de adaptar un equipo EEG convencional, en una herramienta útil tanto para el niño como para el docente. Por ello, el trabajo parte de [51], donde se emplean cuantificadores numéricos de la señal del EEG, provenientes de la metodología de las matemáticas fractales y la teoría de la aproximación. Desde el punto de vista de la aplicación, se buscan las diferencias electroencefalográficas entre los patrones de normalidad y una falta de atención en los niños. Se determina que los niños con TDAH muestran la existencia de una excesiva lenta actividad en el lóbulo frontal.

Por ello, si nos centramos en emplear dos electrodos que cubran el lóbulo frontal del alumno, se pueden determinar diferentes aspectos del trastorno de TDHA y el aprendizaje de este. Esto simplifica mucho la preparación del alumno, permitiendo que sea una herramienta fácil de utilizar en el aula y más recurrente. Así el docente puede asegurar el correcto aprendizaje educativo incluyendo esta herramienta en, por ejemplo, un aula sensorial, si se crea un software que le dé soporte.



### 3 EEG: Bitmed eXea Ultra

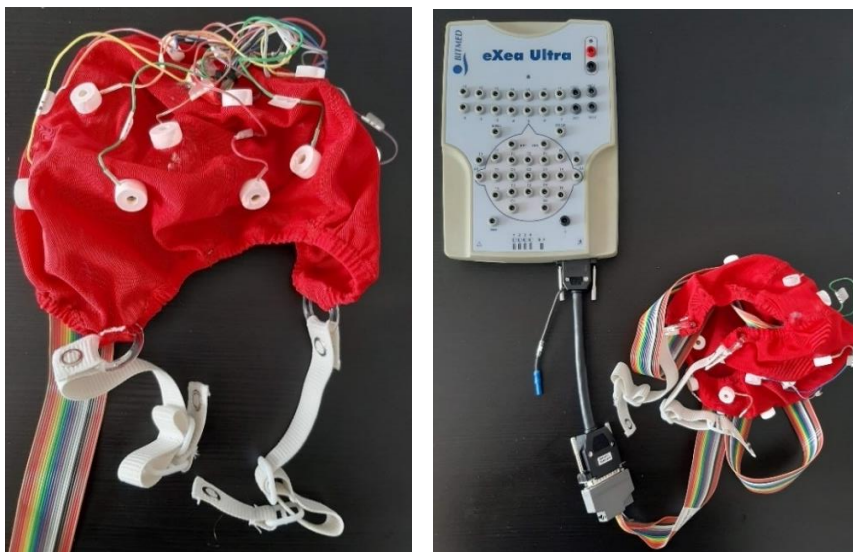
Este apartado contiene una descripción básica del EEG utilizado para este proyecto, desde la perspectiva de su conexión al ordenador y manejo de usabilidad.

El electroencefalograma empleado en el presente TFM se corresponde con el equipo Bitmed eXea Ultra, el cual se muestra en la Ilustración 9.



*Ilustración 9: EEG Bitmed eXea Ultra.*

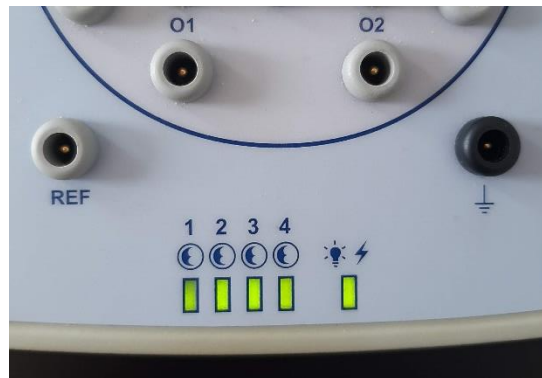
Este registro se toma con un gorro, que tiene integrados los electrodos para reducir el tiempo de preparación del paciente como se puede ver en la Ilustración 10. Para cada uno de los electrodos se tiene que aplicar un gel entre el electrodo y la cabeza del paciente, se ha de rascar previamente el cuero cabelludo. La conexión entre el gorro y la máquina EEG se realiza mediante un adaptador a EEG cap, tal y como se muestra en Ilustración 10.



*Ilustración 10: Electrodo EEG. (a) Gorro de electrodos. (b) Conexión gorro al EEG.*

Posee 32 canales (AC), los cuales muestran los registros de los electrodos conectados al cuero cabelludo. Estos pueden establecerse con una frecuencia de 10, 50, 100, 250 o 500 Hz, según su programación. Por otro lado, hay 7 canales con frecuencia fija, 10 Hz, que se corresponden con: 2 canales DC, ritmo de pulso, saturación de oxígeno, marcador de eventos y sensor de luz.

Tiene 4 leds en la parte inferior, los cuales cambian su estado según el evento, tal y como podemos observar en la Ilustración 11. En el caso de no estar enviando datos, los 4 leds permanecen encendidos en color verde, en el caso de estar enviando datos permanecen parpadeando en color verde, y en el caso de que ocurra algún error se encienden de color rojo. Además, indica la capacidad de memoria, dependiendo del tiempo que se vaya a registrar datos, mediante el color amarillo en cada uno de los leds. Junto a ellos encontramos un indicador de batería, si se está cargando se mantiene permanentemente en color verde, si no se está cargando, pero tiene batería suficiente, parpadea en verde, en caso de quedar poca batería muestra en primer lugar el naranja y en última instancia el color rojo.



*Ilustración 11: Leds indicadores.*

Además, en uno de los lados encontramos otro led indicador de la batería, el cual se puede observar en la Ilustración 12, que se muestra en color verde si en ese preciso momento se está cargando, en otro caso se encuentra apagado.



*Ilustración 12: Led indicador de batería.*

En el costado delantero se pueden observar las entradas para la comunicación con el ordenador y la carga del EEG, tal y como se puede observar en Ilustración 13. Estas conexiones se realizan mediante pines y tienen un mecanismo de seguridad para fijar los cables al EEG.



*Ilustración 13: Entradas EEG.*



Mencionar, que el cable de la alimentación, en las proximidades al EEG, posee una ferrita para evitar interferencias electromagnéticas. La conexión del ordenador con el EEG, para establecer la comunicación serie se realiza mediante el puerto USB del ordenador y la entrada PC del EEG. En la Ilustración 14 se puede observar el tipo de cable y su conexión al EEG.



*Ilustración 14: Cables y conexiones del EEG.*

## 4 Registro del EEG

Este apartado describe el proceso que permite implementar la comunicación serie entre el EEG y el ordenador, la configuración de los comandos y el orden en que se reciben los datos con la explicación de su correspondiente formato.

Para poder establecer la comunicación entre el ordenador y el EEG, inicialmente se tienen que solucionar las incompatibilidades con los drivers. En primer lugar, se abre el EEG para visualizar su circuito interno y se encuentra que la máquina incorpora el chip FT232BL de FTDI, tal y como se puede ver en la Ilustración 15. Este chip, implementa la interfaz entre el puerto USB del ordenador y la UART del microcontrolador.

El problema radica en que, los ficheros (.inf) proporcionados para instalar los drivers están modificados para que al conectar el EEG al puerto USB, el ordenador lo reconozca con el nombre: "Bitmed XGPLab Device", con PID=D7E8. Si se instalan estos drivers directamente el ordenador no es capaz de identificar el EEG por su nombre, para ello se tienen que modificar los ficheros ftdibus.inf y ftdiport.inf añadiendo las líneas de código que incluyen los dispositivos BITMED con dicho PID. También se modifica la descripción para que aparezca como dispositivo USB "Bitmed Device" y como puerto virtual "Bitmed Virtual Port". Éste último no se utiliza, pero se ha de instalar también con un correspondiente orden. Todo esto se encuentra detallado en el manual de instalación, en el *Anexo 5*.

Hay que tener en cuenta que los fabricantes firman digitalmente los drivers para evitar *hackeos*. Al modificar el archivo .inf del driver, su firma deja de ser válida y el sistema operativo Windows 10 no deja instalar drivers con firma inválida salvo que se indique expresamente, para ello hay que deshabilitar el uso obligatorio de controles firmados.

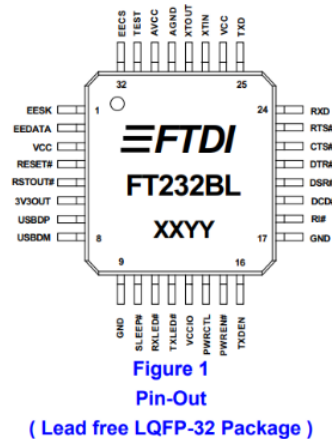


Ilustración 15: Chip FT232BL. (a) Imagen chip FT232BL. (b) Esquemático chip FT232BL .

Una vez resueltas las incompatibilidades con los drivers, hay que establecer la comunicación serie entre el ordenador y el EEG, para ello es imprescindible conocer el baud rate que utiliza este. Gracias a 'ConnTest', una aplicación proporcionada en su día por el fabricante, se pueden intercambiar comandos básicos con el EEG y, por tanto, ver las tramas que entran y salen en la UART con un osciloscopio o con un analizador lógico.

Analizando el datasheet del chip FT232BL, se encuentra que los pines 24 y 25 corresponden con las RDX (recibir datos) y TXD (transmitir datos), respectivamente, de la comunicación asíncrona, tal y como se puede ver en la Ilustración 15. Para poder establecer la comunicación se necesita conocer la velocidad de transmisión. Se conecta el osciloscopio a esos dos pines y se envían comandos con la aplicación 'ConnTest', así se obtiene que el tiempo del bit es de 1µs, por lo tanto, su baud rate es de 1Mbit/s.

Para confirmar que es correcto el baud rate y analizar cómo se envían los comandos al EEG se emplea la herramienta 'SerialPlot'. En primera instancia, se verifica que el programa detecta correctamente el EEG como 'Bitmed Virtual Port' y permite abrir el puerto donde está conectado. A continuación, se introduce el baud rate y se envían algunos comandos simples proporcionados por el fabricante sobre el protocolo empleado por el EEG ExeaUltra en comunicación serie, que se encuentran en el *Anexo 7*.

En la Ilustración 16 se puede ver cómo al enviar los comandos en formato hexadecimal, el EEG los recibe correctamente y comienza a enviar datos al ordenador.

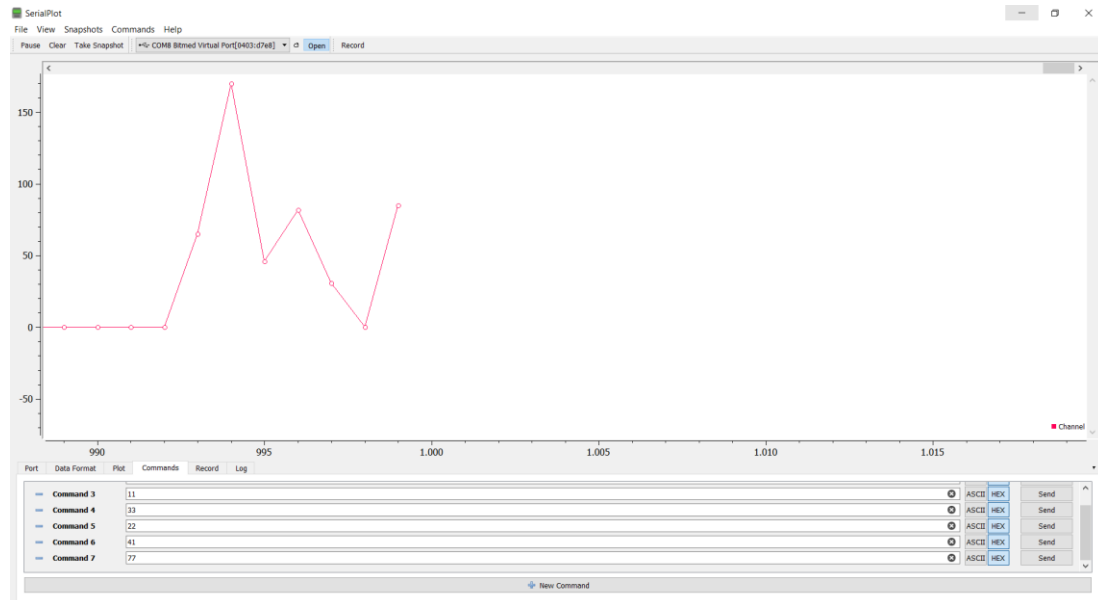


Ilustración 16: Captura de la aplicación SerialPlot, se corresponde con los primeros datos recibidos del EEG.

Así pues, una vez establecida la comunicación, se debe analizar la configuración de los comandos para poder demandar al EEG el envío de una determinada información.

## 4.1 Comandos

La comunicación entre el ordenador y el EEG se establece vía serie, mediante el envío de comandos y la recepción de acuse de recibo (ACK).

Hay unos comandos simples, tales como: 'Borrar la memoria de servicio' (0x22), 'Obtener la configuración del modo holter' (0x69) o 'Obtener la memoria disponible' (0x47). Estos se basan en que, una vez recibido el comando el EEG, este envía el correspondiente ACK y a continuación se empiezan a enviar los datos correspondientes a la petición del comando. Es decir, el comando no se configura. Estos comandos, tal y como se ha comentado antes, son gracias a los que se ha podido establecer el valor del baud rate para la comunicación y para verificar que esta se podía implantar.

El otro tipo de comandos son los que se tienen que calcular según parámetros como la frecuencia o el número de canales. Algunos de estos son: 'Configurar el modo holter' (0x67) o 'Empezar una nueva prueba en tiempo real' (0x11), resultando este el más relevante.

En la Tabla 2 se observa la totalidad de los comandos citados en el *Anexo 7*.

<b>Tarea</b>	<b>Comando</b>
Empezar una prueba en tiempo real	0x11
Parar una prueba en tiempo real	0x17
Borrar la memoria del dispositivo	0x22
Comenzar la descarga de la memoria del dispositivo	0x33
Configurar el dispositivo para el modo holder	0x67
Obtener la configuración del modo holder	0x69
Obtener la memoria disponible	0x47
Obtener la memoria total	0x41
Empezar la medida de impedancias	0x77
Empezar calibración	0x88

*Tabla 2: Comandos EEG.*

#### 4.1.1 Empezar una prueba en tiempo real

En la documentación de partida, la configuración del comando para iniciar una nueva prueba en tiempo real, que se encuentra en el *Anexo 7*, es escueta y escasa. Para poder determinar con más claridad la configuración de este comando, se puede partir del comando simple ‘Obtener la configuración del modo holder’ (0x69) ya que el EEG, tras enviar la comprobación del ACK, envía una de las posibles configuraciones para el funcionamiento de la prueba en tiempo real, pero con algunos datos extra. Esta información se obtiene con la herramienta ‘SerialPlot’.

Tras comparar esta información obtenida con la del protocolo proporcionado y hacer varias deducciones junto a numerosas pruebas, se obtiene una configuración de partida del comando para poder hacer la primera prueba en tiempo real.

Hay que mencionar que no es necesario que los cables de los electrodos estén conectados para poder realizar la prueba. A continuación, se explica la configuración del comando ‘Empezar una nueva prueba en tiempo real’ (0x11) para los 32 canales, en la Tabla 3 encontramos su formato general.

<b>Configuración del comando</b>
Comando
Número de bytes
Configuración de canales
Tasa de muestreo
Módulos del canal
Índice de relación
Tamaño del paquete

*Tabla 3: Configuración del comando 'Empezar una prueba en tiempo real'.*

A continuación, en la Tabla 4 se encuentra un ejemplo detallado de los valores de la configuración para la frecuencia máxima, 500Hz:

Configuración 500Hz, 32 canales
<pre>comnd_ComDirect=[   17, 126, 0, 0, 0, 0, 0, 0, 0, 0,   244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1,   244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1,   244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1,   244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1, 244, 1,   10, 0, 10, 0, 10, 0, 10, 0, 10, 0, 10, 0,   1, 1, 1, 1, 1, 1, 1, 1,   1, 1, 1, 1, 1, 1, 1, 1,   1, 1, 1, 1, 1, 1, 1, 1,   1, 1, 1, 1, 1, 1, 1, 1,   50, 50, 50, 50, 50, 50,   50, 0,   140, 12 ]</pre>

Tabla 4: Comando 'Empezar una prueba en tiempo real', 500 Hz, 32 canales.

Se procede a explicar la procedencia de cada uno de los valores que configuran este comando, en formato decimal:

**17:** Se corresponde con el comando 0x11 en decimal, 'Empezar una nueva prueba en tiempo real'.

**126:** Refleja el número de bytes, se calcula tal y como se muestra en la Ecuación I, los valores correspondientes se observan en la Ecuación II.

*Ecuación I*

$$N = \frac{n^{\circ}\text{canales AC}}{4} + 2 * n^{\circ}\text{canales} + n^{\circ}\text{canales} + 2 + 2$$

*Ecuación II*

$$126 = \frac{32}{4} + (2 * 32 + 6 * 2) + (32 + 6) + 2 + 2$$

Los canales AC (nº de canales AC) son 32, mientras que todos los canales (nº de canales) son los 6 canales de frecuencia fija y los 32 AC.

Los siguientes números se corresponden con los bytes adicionales y se han de incluir en el siguiente orden:

**0:** Configuración de canales, se envían 'N' bytes de valor 0. El cálculo de 'N' se corresponde con la Ecuación III.

*Ecuación III*

$$N = \frac{n^{\circ}\text{canales AC}}{4} = \frac{32}{4} = 8$$

Por lo que, para configurar este parámetro, se envía el valor 0, 8 veces separados por comas.

**244, 1 y 10, 0:** Tasa de muestreo, se corresponde con introducir el valor de la frecuencia en 2 bytes en el siguiente orden: Canales AC, ritmo de pulso, saturación de oxígeno; canal DC1 y canal DC2, sensor de luz y marcador de eventos.

Los bytes se transmiten en formato 'little endian', comenzando la transmisión por el byte menos significativo.

- **244, 1**-> En primer lugar se calcula el valor de los 32 canales AC para los cuales se ha seleccionado un valor de 500 Hz. Como a cada uno le corresponden 2 bytes, se tienen 64 bytes. El número decimal 500 no cabe en un byte (ya que incluye de 0 a 256), por lo que, en este caso, emplearemos los dos bytes tal y como se observa en la Ecuación IV.

*Ecuación IV*

$$500 \text{ (decimal)} = 0x1F4 \text{ (hexadecimal)}$$

El número decimal 500 se corresponde con el 0x1F4 en hexadecimal, la transmisión comienza por el byte menos significativo, quedando en decimal de la siguiente forma: 244,1.

- **10,0**-> Este valor se corresponde con los restantes 6 canales (ritmo de pulso, saturación de oxígeno, canal DC1, canal DC2, sensor de luz y marcador de eventos). Para cada uno corresponden 2 bytes, por lo que la configuración constará de 12 bytes y todos tienen una frecuencia fija de 10 Hz.

**1 y 50:** Módulos del canal. Le corresponden 1 byte por canal, primero los canales AC y luego los canales de frecuencia fija, se calcula mediante la Ecuación V.

*Ecuación V*

$$N = \frac{500}{\text{Tasa de muestreo}}$$

- **1**-> Para los canales AC hay 32 bytes. El resultado es el valor de la Ecuación VI, separado por comas.

*Ecuación VI*

$$N = \frac{500}{\text{Tasa de muestreo}} = \frac{500}{500} = 1$$

- **50** -> Para el resto de los canales que se corresponden con los de frecuencia fija, hay 6 bytes, con el resultado de Ecuación VII separado por comas.

*Ecuación VII*

$$N = \frac{500}{\text{Tasa de muestreo}} = \frac{500}{10} = 50$$

**50,0:** Índice de relación. Le corresponden 2 bytes en formato 'little endian', con el resultado de Ecuación VIII.

*Ecuación VIII*

$$N = \frac{\text{Máxima tasa de muestreo}}{\text{Mínima tasa de muestreo}} = \frac{500}{10} = 50$$

**140, 12:** Tamaño del paquete. Le corresponden 2 bytes en formato 'little endian', con el resultado de la Ecuación IX.

*Ecuación IX*

$$N = 2 + a + 10$$

Correspondiéndose su valor con el de la Ecuación X.

*Ecuación X*

$$a = 2 * \frac{\sum(\text{tasa de muestreo de cada canal})}{10}$$

Tras varias pruebas se concreta que el sumatorio se corresponde solo a los canales AC, quedando el resultado de la Ecuación XI.

*Ecuación XI*

$$a = 2 * \frac{500 * 32}{10}$$

Por lo tanto, el resultado final se muestra en la Ecuación XII.

*Ecuación XII*

$$N = 2 + 2 * \frac{500 * 32}{10} + 10 = 3212$$

El número decimal 3212 se corresponde con el 0xC8C en hexadecimal, la transmisión comienza por el byte menos significativo, quedando en decimal de la siguiente forma: 140, 12.

Hay que mencionar que los siguientes canales de frecuencia fija: ritmo de pulso, saturación de oxígeno, canal DC1, canal DC2 están formados por 2 bytes, mientras que los canales correspondientes con sensor de luz y marcador de eventos poseen 1 byte.

Después de enviar dicho comando al EEG, este responde con un ACK compuesto por dos bytes, uno fijo: 0xFD y otro que depende de la máquina que se esté empleando, que, en el caso de Bitmed eXea Ultra, corresponde con el 0x08.

Una vez recibido el ACK, el EEG envía 10 paquetes por segundo, estando cada paquete formado por 3212 bytes en total, los cuales se corresponden con:

- 2 bytes de preámbulo.
- 10 bytes de los canales de frecuencia fija (2 bytes para los 4 primeros y 1 byte para los dos últimos).
- 3200 bytes de los 32 canales AC. Cada canal aporta 100 bytes que se corresponden con 50 muestras, ya que son de 2 bytes. Hay 50 muestras tal y como indica el índice de relación ya que por cada valor de los canales de frecuencia fija se obtienen 50 de cada canal AC.

Para la aplicación del EEG en el ámbito de la educación no es necesario que los canales tengan diferentes frecuencias, por lo que, en nuestro caso, los 32 canales se configuran a la misma frecuencia: 10, 50, 100, 250 o 500 Hz. La configuración para el resto de las frecuencias se encuentra en el *Anexo 2*.

Se verifica que el comando se ha enviado de forma correcta cuando, tras enviarlo: los leds parpadean en verde, el ordenador recibe los ACK y comienza el registro de datos.

Para finalizar la prueba en tiempo real se ha de enviar el comando 'Finalizar una prueba en tiempo real' (0x17), sin argumentos. El EEG envía como ACK ese mismo comando y finaliza la emisión de datos, pero hay que tener en cuenta que, después de enviar el comando, puede haber datos disponibles en el buffer de entrada y se tienen que leer antes para comprobar el ACK.

## 4.2 Proceso de recogida de datos

En primer lugar, gracias a la biblioteca 'Serial' de python se enumeran los puertos serie disponibles en el ordenador (esta biblioteca se encuentra más detallada en el *Anexo 1*). Para el caso del EEG, por ejemplo, si se conecta en el puerto número cuatro, lo detecta de la siguiente forma:

COM4 Bitmed Virtual Port (COM4) [USB VIDPID=0403D7E8 SER=5]

Las características detalladas de este ejemplo se encuentran en la Tabla 5.

Característica	Valor
Puerto	4
Descripción asociada	Bitmed Virtual Port
Hardware ID	[USB VIDPID=0403D7E8 SER=5]

*Tabla 5: Características puerto serie.*

Una vez se ha detectado el EEG por el ordenador, este lo establece automáticamente como el puerto con el cual se implantará la comunicación. Para realizar la comunicación es necesario conocer el valor del baud rate determinado en el apartado Registro del EEG de esta memoria. De esta forma se queda configurado y abierto el puerto serie correspondiente al EEG.

A continuación, antes de que el EEG envíe datos, se verifica que tanto el buffer de entrada como de salida del puerto no contienen datos de otra ejecución. Se establece una determinada configuración del comando ‘Empezar una nueva prueba en tiempo real’ (0x11), dependiendo de la frecuencia, tal y como se ha mencionado en el apartado anterior, y se envía al EEG.

Si la configuración del comando es correcta, los leds del EEG se iluminarán parpadeando en color verde y enviará en primer lugar el ACK. El ACK es comprobado por el ordenador dato a dato.

A partir de aquí, en bucle, el EEG envía diez paquetes de datos por segundo, de los cuales se recibe primero el preámbulo (el cual también se comprueba dato a dato), y después los datos de los canales de frecuencia fija y los datos de los 32 canales.

Centrándonos en los datos de los 32 canales, hay que leer del puerto serie los bytes que forman las muestras correspondientes a cada canal, para cada frecuencia. Cada muestra está constituida por dos bytes en formato ‘little endian’, la operación para convertirlos en un entero se realiza mediante el método preconstruido en python: `int.from_bytes`.

Las muestras se almacenan en listas organizadas por canales para su posterior guardado en ficheros de texto, con formato ‘.txt’. Finalmente, cuando no queda ninguna operación por realizar se envía el comando ‘Finalizar una prueba en tiempo real’ (0x17) y se cierra de forma segura el puerto serie elegido para dar por terminada la comunicación con el EEG.



# 5 Procesado de los datos EEG y su representación gráfica

Este apartado describe qué gráficos se han implementado en este proyecto y qué información aportan al experto en cuestión. Se busca representar los datos obtenidos del EEG, en primer lugar, mediante una visualización 2D donde se muestre su evolución. En segundo lugar, se realiza la FFT de los datos para que estos puedan representar valores pedagógicamente útiles, se realizan los gráficos: diagrama de sectores y gráfico de barras. Finalmente, para ofrecer una visión más general del proceso, se implementa un mapa topográfico, es decir una cartografía cerebral.

## 5.1 Visualización 2D

Busca representar, para los canales seleccionados, un gráfico de dos dimensiones donde el eje ‘x’ se corresponda con el tiempo y el eje ‘y’ con los datos obtenidos del EEG.

Se realiza usando la biblioteca de gráficos científicos ‘PyQtGraph’, la cual destaca por ser muy rápida (en el *Anexo 1*), se encuentra una descripción más detallada. Se descarta emplear la biblioteca ‘Matplotlib’ para realizar este gráfico porque, a pesar de ser más madura y sencilla, no es apta para gráficos que precisen actualizaciones rápidas de trama y video [52].

Para esta representación, se crea un temporizador de tiempo cero que llama repetidamente a una método `update()` cada vez que este termine de realizar su código. En dicho código se incluye la lectura de cada paquete, separándola por canales con sus correspondientes comprobaciones y tratamiento, además, al final de este, se encuentra la llamada a un segundo método. Este segundo método se crea para que actualice el dato y la posición de los gráficos, es decir, se encarga de representar los paquetes de los canales seleccionados por el usuario. El resultado se puede observar en la Ilustración 17, para un solo canal representado a una frecuencia de 500Hz.

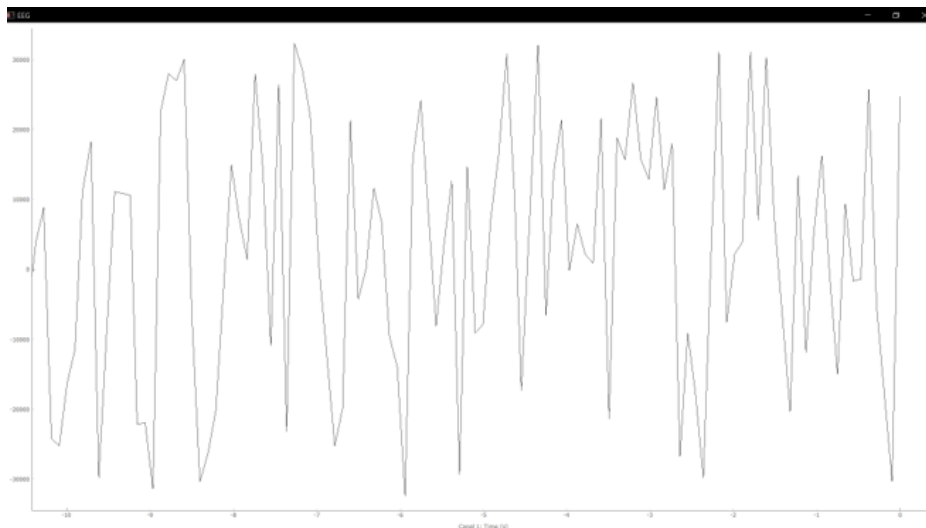
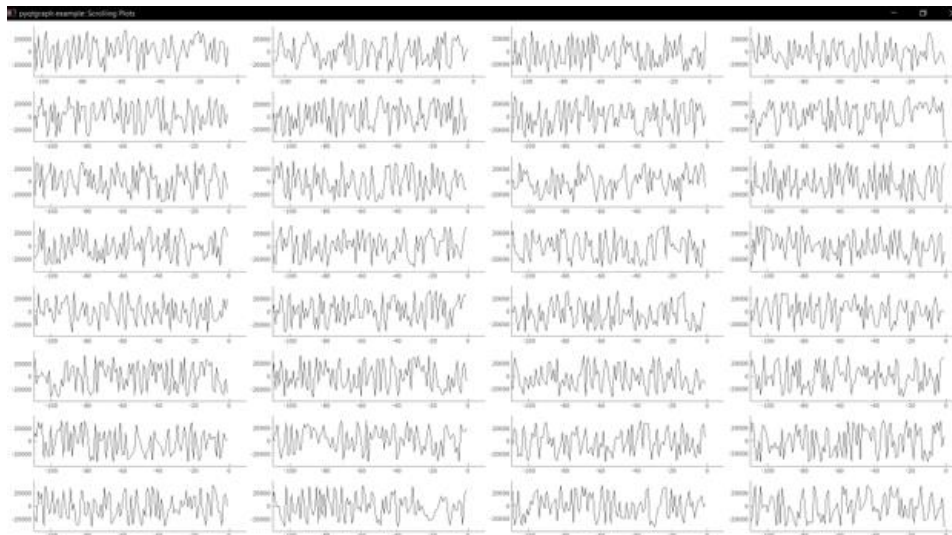


Ilustración 17: Visualización 2D de un canal, 500Hz.

Previo a esto, se ha establecido que el máximo número de gráficos por cada fila es cuatro, es decir, como hay 32 canales, como máximo habrá ocho filas, tal y como se puede ver un ejemplo de todos los canales a una frecuencia de 500 Hz en la Ilustración 18. Así pues, cuando el usuario introduce el número de canales que quiere, estos se distribuyen rellenando y completando filas de cuatro gráficos, creando un subplot para cada canal.



*Ilustración 18: Visualización 2D de todos los canales, 500Hz.*

Seguidamente, al usuario le aparece una ventana emergente en la que, en tiempo real, se representan los nuevos datos, los cuales se desplazan de derecha a izquierda. Es decir, los datos actualizados se encuentran en valor 0 del eje de la 'x'. Además, existe la posibilidad de mover los datos representados arrastrando el gráfico con el ratón en la dirección deseada, con esto se permite al usuario analizar un evento concreto.

Respecto al número de canales que se pueden representar en tiempo real, hay que tener en cuenta que, cuantos más canales se representen, más tiempo de cómputo se precisa para representarlos, por lo que, llegado a un número de canales, el sistema se colapsa y paraliza algunos gráficos. También interfieren los procesos que se estén realizando de forma paralela en el ordenador, llegando a no trabajar en tiempo real dependiendo de ambos factores mencionados. Hay que tener en cuenta que, aparte de esta problemática, el representar los 32 canales a la vez carece de sentido, ya que el usuario no es capaz de analizar tan rápido tantos canales.

## 5.2 FFT

La parte más importante de la cartografía cerebral es poder determinar las características de la actividad recogida en cada electrodo y verificar si se encuentra dentro de los límites normales para cada paciente y situación. Así, una vez digitalizada la señal biométrica se debe procesar matemáticamente para evaluar parámetros como, por ejemplo, la potencia absoluta y relativa de las diferentes bandas de frecuencia, ya que dan información sobre las características y el comportamiento de la actividad bioeléctrica cerebral. La potencia absoluta es un índice de la cantidad de energía existente en una banda de frecuencia determinada y en un punto concreto de la superficie craneal. La potencia relativa indica el porcentaje de actividad de una banda de frecuencia y su distribución topográfica. Describen aspectos de la actividad bioeléctrica cerebral y son sensibles a los cambios psicofísicos del sujeto y a la presencia de diferentes enfermedades [53].

El análisis espectral se realiza con el método de la densidad espectral, en nuestro caso nos vamos a centrar en la potencia relativa. Las señales del EEG para cada canal se dividen en secuencias cortas y la transformada de Fourier se halla en esas secuencias cortas con el cálculo de la magnitud del resultado al cuadrado [54].

Se puede obtener la potencia espectral de una secuencia de datos en python con ayuda de las funciones de la biblioteca 'NumPy' y, gracias a la creación de un diccionario, los resultados obtenidos se pueden dividir entre las diferentes bandas de frecuencias. El resultado, de tomar datos del EEG y aplicar estos procedimientos se puede observar en la Ilustración 19.

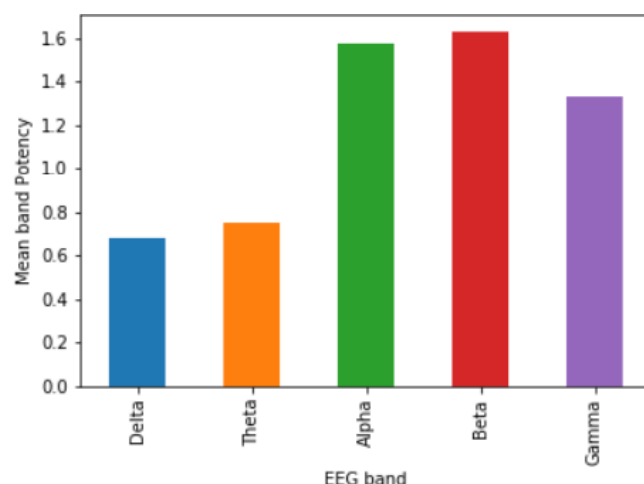


Ilustración 19: Ejemplo de la distribución de los datos sobre las bandas de frecuencia.

Hay 5 bandas clínicas estándar para señales de EEG conocidas como: Delta (0,1 a 4 Hz), Theta (4 a 8 Hz), Alfa (8 a 13 Hz), Beta (14 a 30 Hz) y Gamma (> 30 Hz) [8]. En nuestro caso se emplea la potencia relativa, es decir la potencia de cada banda respecto a la suma de todos los valores de potencia obtenidos. En el *Anexo 3* se describe como se verifica el funcionamiento del código implementado.

De esta forma, los datos del EEG se pueden transformar, haciendo su FFT, y se pueden dividir en las distintas bandas de frecuencia, convirtiéndose en una fuente potente de información para poder realizar un análisis de la actividad cerebral. Una vez realizada esta división, se precisa plasmarlo gráficamente para facilitar el análisis de los datos. A continuación, se detalla la implementación de los datos en un gráfico de barras y un diagrama de sectores.

### 5.2.1 Diagrama de sectores y gráfico de barras

Se busca representar las diferentes bandas de frecuencia de los datos extraídos del EEG en un diagrama de sectores y en un gráfico de barras. Para ambas se emplea la biblioteca 'Matplotlib', la cual se encuentra detallada brevemente en el *Anexo 1*. Con esto se pretende facilitar el análisis visual de los datos por parte del experto, ya que en caso de que alguna de las bandas de frecuencias tenga un valor extremo, este se podrá detectar rápidamente.

Para realizar la representación, en primer lugar, se leen los datos del EEG en bruto, se hace la FFT, se organizan por bandas de frecuencia y finalmente se generan tantas subfiguras como sean requeridas. Cada una de las subfiguras representadas consta de 100 datos brutos, ya que se considera que ese valor de muestra es suficientemente representativo para el gráfico.

Para este tipo de gráficos se tiene que crear una figura inicial, ya que, como los datos se leen en tiempo real, hay que esperar a tener cien datos de los canales necesarios para poder tratarlos y posteriormente representarlos. Por ello, inicialmente, se muestran las subfiguras en blanco, solo con los ejes y nombres, y luego cuando se poseen los datos se actualiza.

En el apartado de aplicación, para darle versatilidad se explica la inclusión de un 'slider' (barra deslizadora) en uno de los modos de funcionamiento del EEG y su actualización sobre el mismo gráfico.

Se puede observar el resultado de cada uno de los gráficos en la Ilustración 20. Si se observa con detenimiento, los valores, porcentajes y colores de los dos gráficos, para el mismo canal y para la misma frecuencia coinciden.

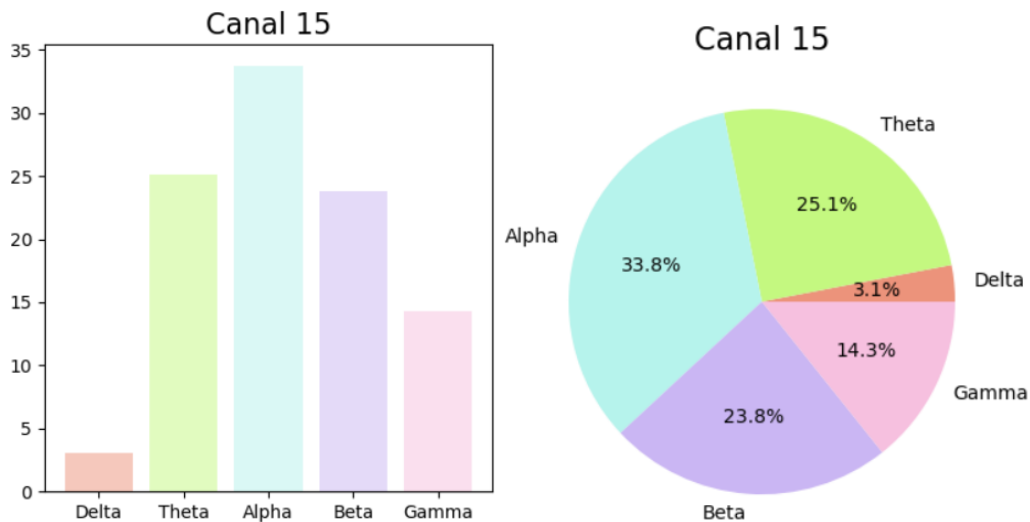


Ilustración 20: Gráficos FFT. (a) Gráfico de barras. (b) Diagrama de sectores.

### 5.3 Cartografía cerebral

Se busca realizar un mapa topográfico con todos los canales del EEG, para ello se emplea el paquete MNE, detallado en el *Anexo 1*.

La cartografía cerebral se basa en considerar la posición de varios electrodos, en nuestro caso los 32 que posee la herramienta EEG, para interpolar los datos obtenidos del EEG y disponer de una distribución visual de estos en forma espacial sobre un dibujo de la cabeza del paciente.

Es importante mencionar que la presencia de datos en zonas específicas tiene significancia pedagógica y emocional. Por ejemplo, debemos tener en cuenta que la facultad de reconocimiento de caras está regida por regiones cerebrales situadas en las caras inferiores de los lóbulos temporal y occipital, a ambos lados de la corteza. Los mapas cerebrales espectrales ayudan, de hecho, a establecer una diagnosis de maduración cerebral, no accesible hasta ahora mediante la mera inspección visual del registro EEG [55].

Se representan los datos de los 32 canales en diferentes colores acordes a su leyenda en  $\mu V$  y el tiempo junto al título del gráfico tal y como se puede ver en Ilustración 21. En el apartado de aplicación se explica su implementación desde archivo con un 'slider' (barra deslizadora).

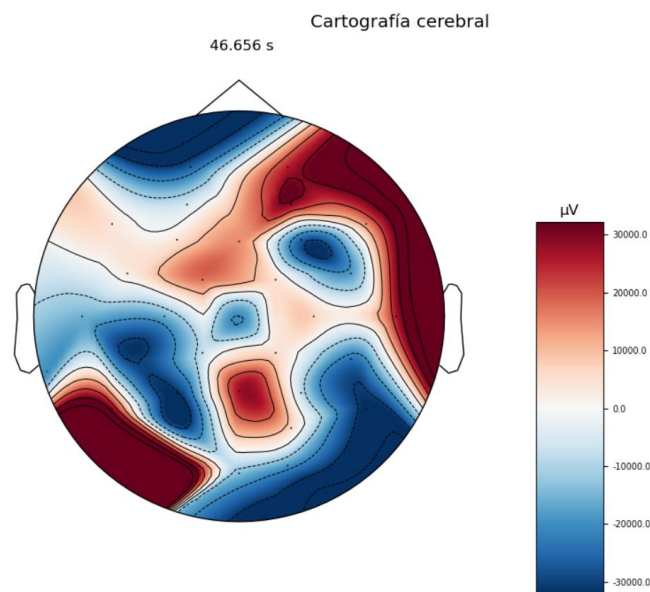


Ilustración 21: Ejemplo cartografía cerebral.

# 6 Aplicación

En este apartado se describen los requisitos y características de la aplicación: se realiza primero un análisis sobre sus usos y opcionalidades, a continuación, se concretan aspectos de diseño y finalmente se explica su implementación.

El objetivo principal de desarrollar esta aplicación es dotar de interfaz de usuario al código ya implementado para que se pueda emplear en el ámbito educativo, además de integrarlo de forma robusta, buscando una herramienta clara e intuitiva que permita al experto obtener un análisis detallado del registro de un paciente, en nuestro caso un niño con TDAH.

## 6.1 Análisis

Se realiza un análisis sobre la aplicación, sus usos, sus opcionalidades, etcétera. Hay que mencionar que tanto los requisitos funcionales como los no funcionales se establecen en la fase inicial del diseño de la aplicación.

### 6.1.1 Modos

La aplicación posee dos modalidades de uso:

- **Nuevo registro:** Permite realizar un registro en tiempo real obteniendo los datos del EEG conectado al ordenador por el puerto serie. Se permite guardar esos datos directamente o verlos en diferentes gráficos conforme se actualizan.
- **Registro de archivo:** Se parte de un registro realizado previamente, este ha de estar guardado en el formato adecuado. Se permite visualizar los datos tanto en los diferentes gráficos como en forma tabulada.

Los gráficos implementados son los mencionados en el apartado anterior: gráfico de barras (bandas de frecuencia), diagrama de sectores (bandas de frecuencia), visualización 2D y cartografía cerebral (solo en el modo registro de archivo).

### 6.1.2 Requisitos funcionales

A continuación, se detallan los requisitos funcionales de la aplicación, los cuales representan los servicios que prestará dicho sistema y sus opcionalidades.

1. La aplicación permitirá al usuario seleccionar entre realizar un nuevo registro en tiempo real o consultar un registro de archivo.
2. La aplicación será dinámica, permitirá al usuario navegar entre las diferentes vistas permitiendo retroceder a la vista anterior en el caso de que así lo desee.
3. La aplicación permitirá realizar con los datos del registro, ya sea nuevo o de archivo, los siguientes gráficos en relación con la FFT: gráfico de barras y diagrama de sectores.
4. La aplicación permitirá realizar con los datos del registro, ya sea nuevo o de archivo, realizar una visualización 2D de los datos.
5. La aplicación permitirá realizar con los datos de un registro de archivo un mapa topológico, es decir, una cartografía cerebral de los 32 canales.
6. La aplicación, en el caso de que el usuario seleccione un nuevo registro, permitirá que se guarden los datos sin ninguna visualización.
7. La aplicación, en el caso de que el usuario seleccione cargar un registro de archivo, permitirá visualizar en pantalla de forma tabulada los datos que se empleen para el gráfico seleccionado.
8. La aplicación, en el caso de que el usuario seleccione realizar un nuevo registro, permitirá guardar los datos de dicho registro para que puedan ser cargados posteriormente en la opción: cargar un registro de archivo.

9. Todas las opciones vendrán con su debida descripción para facilitar el entendimiento del usuario.
10. La aplicación tendrá entradas para que el usuario seleccione los canales y frecuencias deseados.
11. El número de canales implementados para los gráficos será dos, por los motivos educativos mencionados anteriormente, menos para la cartografía cerebral donde se emplearán todos los canales.
12. En caso de que los parámetros introducidos en los diferentes campos sean incorrectos, la aplicación informará al usuario de dicho error, permitiendo que este regrese a la vista anterior.

### 6.1.3 Requisitos no funcionales

Los requisitos no funcionales hacen referencia a las propiedades y restricciones del sistema.

1. La interfaz desarrollada con sus correspondientes textos descriptivos permitirá la fácil navegación por las vistas implementadas.
2. La instalación de la aplicación en su totalidad se realizará en menos de media hora.
3. La aplicación será de fácil usabilidad, el usuario podrá conocer los dos modos de funcionamiento en menos de cinco minutos.
4. Las funcionalidades del sistema podrán ser aprendidas en menos de media hora por un usuario sin previa experiencia con la aplicación.
5. El tiempo de respuesta de la aplicación en cualquiera de sus modos y opciones no sobrepasará los cuarenta segundos.
6. Los datos de un nuevo registro se guardarán en diferentes ficheros en formato texto.
7. Los datos empleados en un registro de archivo se visualizarán de forma tabulada, donde se detallarán el número de los canales y el valor correspondiente de cada dato.
8. Se facilitará las entradas al usuario mediante: entradas de texto, radio buttons y data-list.
9. El mantenimiento de la aplicación se podrá realizar de forma eficiente con la documentación del manual de usuario.
10. En el caso en el que el usuario seleccione un nuevo registro, el EEG deberá estar conectado al puerto serie del ordenador, cargado y encendido.
11. Se precisará de un mínimo de datos, en el modo registro de archivo, para poder visualizar los diferentes gráficos.
12. En el modo registro de archivo se facilitará la visualización de los datos mediante un slider.
13. Se podrá extraer datos de un solo EEG con esta aplicación de forma simultánea.

## 6.2 Diseño

A continuación, se explica el proceso de diseño de la aplicación: la arquitectura y los bocetos iniciales para su posterior implementación. La estructura de esta se basa en utilizar un servidor web que soporta diferentes ‘endpoints’, estos se corresponden con diferentes rutas específicas que dotan de utilidad a la aplicación. La aplicación, es decir, el servidor web, se encuentra en el ordenador de forma local en la URL: 127.0.0.1:5000 (host: 127.0.0.1 [local-host], puerto: 5000).

### 6.2.1 Arquitectura

Para facilitar la estructura de la aplicación se emplea el framework ‘Flask’, el cual está concebido bajo el patrón arquitectónico MVC (Modelo Vista Controlador). El modelo MVC se basa en separar el código en tres capas diferentes: Modelos, Vistas y Controladores. El controlador es el encargado de modificar al modelo y de actualizar la vista.

A continuación, se describe cada una de las capas mencionadas de forma detallada [56]:

- **La capa Controlador:** se corresponde con el servidor, sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar

las necesidades de la aplicación. Se corresponde con los archivos desarrollados en python.

- **La capa Modelo:** es la capa que trabaja con los datos, en nuestro caso, se comunica con el EEG y accede a los ficheros de datos. Contiene el núcleo de datos y su funcionamiento independientemente del comportamiento de entrada y salida.
- **La capa Vista:** contiene el código que va a producir la visualización de las interfaces de usuario, o sea, el código que permitirá renderizar los estados de nuestra aplicación en HTML. En nuestro caso, gracias al CSS y el framework CSS Bootstrap, se le dota de estilo a los documentos del HTML y mediante JavaScript se le dota de funcionalidad. En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. La vista corresponde con los ficheros HTML de la herramienta.

Como es un servidor web, las vistas se alternan dependiendo de los distintos endpoints que se basan en rutas dentro de la página. El usuario, conforme navega e introduce entradas va modificando la ruta de la siguiente manera:

- **Directorio raíz:** Se corresponde con la vista ‘vista\_inicial’, se muestra los dos posibles modos de funcionamiento de la aplicación. Se selecciona mediante un radio button, independientemente del modo elegido la ruta se modifica al siguiente endpoint /Seleccion.  
La ruta se corresponde con: ‘/’.
- **Selección:** Se corresponde con la vista ‘vista\_serie’ en caso de que el usuario seleccione un nuevo registro y con ‘vista\_archivo’ si elige registro de archivo. En ambas vistas, mediante una data-list se muestran los posibles gráficos en los que representar los datos y para cada uno de ellos hay una entrada de texto con la que seleccionar los canales y una data-list para elegir la frecuencia. Una vez elegido lo anterior, hay un botón de tipo submit con el texto ‘Mostrar’ que ejecuta la acción reclamada por el usuario. Además, hay un radio button en ambas opciones que, en el caso de nuevo registro, permite guardar los datos, mientras que en el caso de registro de archivo permite visualizar datos de forma tabulada.  
La ruta se corresponde con: ‘/Seleccion’.
- **Gráficos:** Este end point ofrece información a partir de las entradas del usuario. Se emplean ‘query strings’ de la forma: ‘?parameter1=value1&parameter2=value2...’, de esta manera se pueden enviar parámetros a la página web al realizar una petición GET. Por lo que, cuando estamos en el endpoint ‘/Selección’, si se elige un tipo de gráfico, o se indican los valores de frecuencia-canales, o si se quiere guardar o visualizar, la aplicación se encarga de capturar estos parámetros. Al pulsar el botón Mostrar no se accede a una nueva vista, sino que se queda en la misma y se modifica la ruta de la ya existente añadiendo los parámetros introducidos por el usuario.  
Por ejemplo, en el caso de marcar registro de archivo y seleccionar el gráfico diagrama de sectores, con frecuencia 20 Hz, canales 0 y 2 y no requerir la opción de visualización, la ruta quedaría de la siguiente manera:  
‘/Seleccion?Sectores\_Frecuencia\_ID=20&Sectores\_Canales\_ID=0%2C1&visualizar=down’.

Además, en caso de que el usuario quiera volver a la vista anterior, hay un botón con el texto ‘Atrás’ que le permite regresar modificando la ruta mencionada.

Por otro lado, en el caso de que el usuario introduzca mal los parámetros requeridos le aparecerá la vista ‘vista\_error’ donde se le indica que hay un error y que puede consultar la documentación para solucionarlo. En este caso la ruta se mantiene sin añadir ningún parámetro.



### 6.2.2 Diagramas de flujo

A continuación, se exponen los diagramas de flujo que explican el funcionamiento de las diferentes partes de la aplicación. En primer lugar, tenemos un diagrama del lanzamiento de la aplicación y un diagrama con la estructura de la aplicación. El código está modulado en lo que respecta a los diferentes tipos de gráficos, por ello cada uno de ellos tiene su propio diagrama.

- **Diagrama del lanzamiento de la aplicación:** En la Ilustración 22 se observa el funcionamiento interno asociado al archivo bat encargado de lanzar la aplicación. Se basa en crear dos hilos de ejecución, uno se encarga de lanzar la aplicación mientras que el otro hilo, en primer lugar, entra en espera y tras unos segundos abre el navegador para que el usuario pueda ver la aplicación. Este diagrama cubre hasta que se abre la aplicación y ya está preparada para ser usada por el usuario, lo que se corresponde con el 'Fin' del diagrama.

Previo a esto, en caso de ser necesario, se instalan todas las librerías suficientes para que funcione el código de la aplicación. Lo relacionado con su lanzamiento se explica posteriormente en la memoria y en *Anexo 5*.

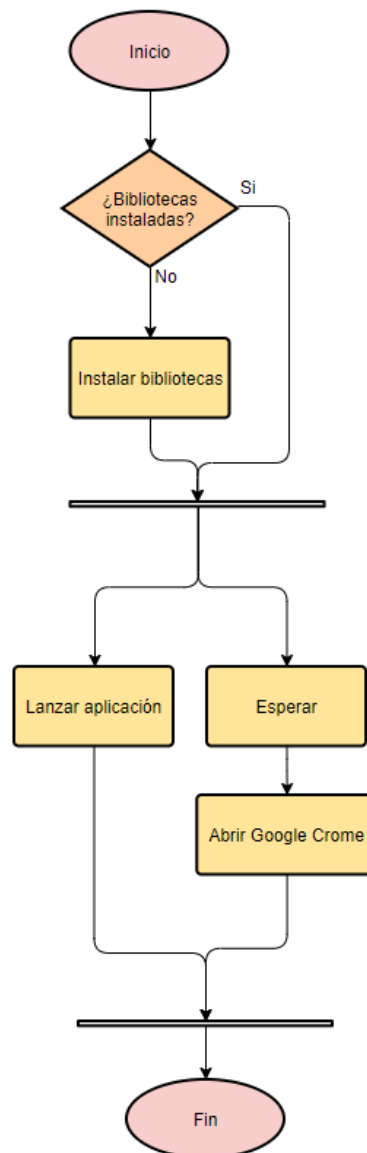


Ilustración 22: Diagrama de flujo de lanzamiento de la aplicación.



➤ **Diagrama estructura de la aplicación:** En la Ilustración 23 se muestra la estructura interna de la aplicación una vez está lanzada, es decir, después del diagrama anterior. En primer lugar, el usuario introduce por pantalla, mediante un radio button, el modo que desea, siendo las dos opciones: Serie y Archivo, las cuales se corresponden con ‘nuevo registro’ y ‘registro de archivo’. Las características de los dos modos ya se han explicado anteriormente y se detallan más en capítulos posteriores.

Si el usuario selecciona realizar un registro de archivo, es decir, el modo archivo, a continuación, ha de introducir el tipo de gráfico que desea y sus parámetros. La aplicación captura esos parámetros y abre dos hilos, uno se encarga de lanzar el gráfico en una nueva ventana y el otro de mantener activa la aplicación, a la vez que, si el usuario ha seleccionado la opción de visualizado de datos, mostrarlos en la ventana de la aplicación. Esto se realiza con hilos independientes para poder visualizar varios gráficos a la vez y obtener así un mejor análisis.

Por otro lado, si se selecciona el modo serie, que se corresponde con iniciar un nuevo registro, igual que en el caso anterior, la aplicación captura los parámetros introducidos por el usuario. En este caso, como la aplicación recibe datos del EEG en tiempo real, una vez que se ha seleccionado un gráfico no se puede acceder a otro hasta que se cierre la ventana y se establezca una nueva comunicación con el EEG, ya que el código se ha modulado en diferentes archivos. Por ello, al crear los dos hilos, el que lanza la ventana con el gráfico no permite una nueva captura de parámetros hasta que el usuario haya cerrado la ventana, mientras tanto el otro hilo mantiene abierta la aplicación.

Se puede volver atrás para cambiar el modo en el que se encuentra la aplicación, por ello el diagrama es bidireccional en lo que se refiere a seleccionar el modo de funcionamiento.

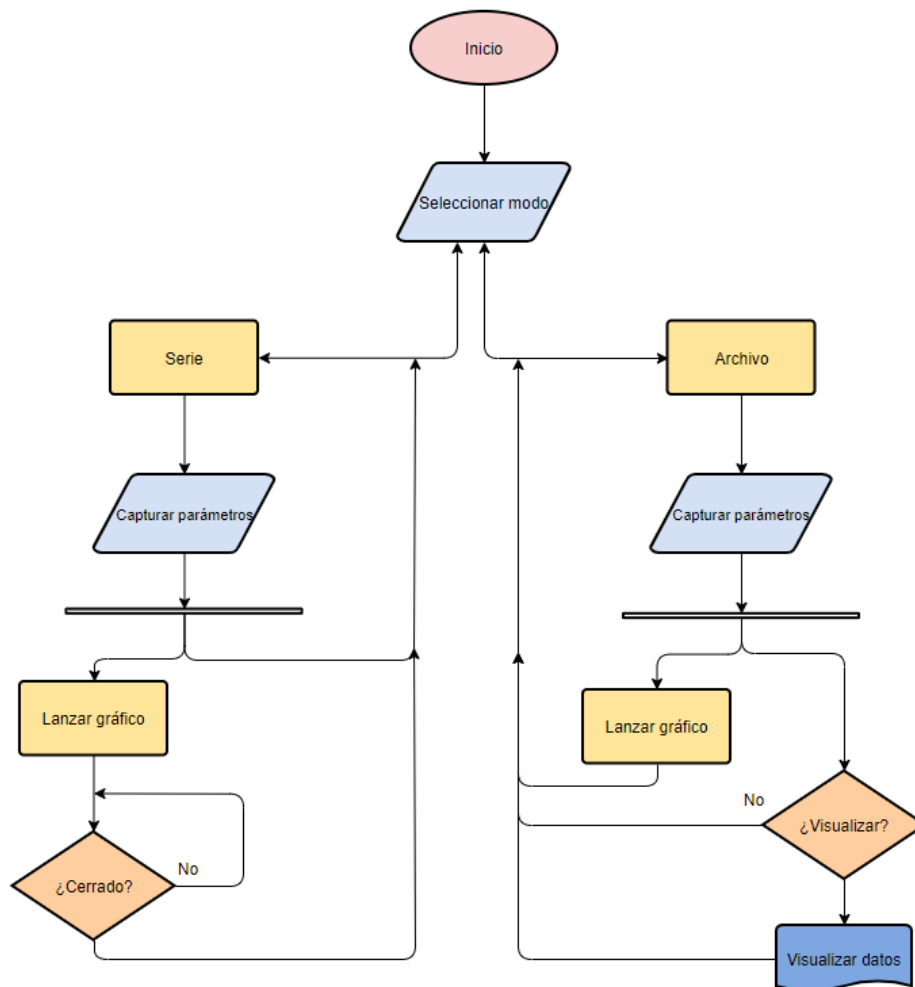


Ilustración 23: Diagrama de flujo de la aplicación.

Como se ha comentado al principio de este punto, cada tipo de gráfico posee un funcionamiento diferente y por tanto diferente diagrama. Los diagramas relativos a los gráficos los encontramos en el *Anexo 4*:

- Diagrama del gráfico de barras y diagrama de sectores del modo serie.
- Diagrama del modo guardado de datos del modo serie.
- Diagrama del modo visualización 2D del modo serie.
- Diagrama del gráfico de barras y diagrama de sectores del modo archivo.
- Diagrama del modo cartografía cerebral y modo visualización 2D del modo archivo.

### 6.2.3 Bocetos

Se adjuntan imágenes del diseño inicial, a mano alzada, de las diferentes vistas de la aplicación. Se recuerda, que es un boceto, es decir, está simplificado, por lo que no está definido a la perfección respecto al aspecto final de la aplicación.



*Ilustración 24: Boceto vista inicio.*

La vista inicio, se corresponde con el directorio raíz, tal y como podemos ver en la Ilustración 24, como título se encuentra el nombre del EEG y la página está dividida en dos partes que se corresponden con los modos de funcionamiento.

A la izquierda se encuentra 'nuevo registro', tal y como se indica en el título de esa subsección, y contiene una descripción de dicho modo. A la derecha se encuentra lo mismo, pero con el modo 'registro de archivo'. Gracias a un radio button se puede elegir uno de estos modos pulsando el botón: Continuar...

Para dotar a la página de más estilo se introducen líneas que delimiten el espacio.

En la Ilustración 25 se puede observar el boceto de la vista archivo, que consta de dos partes diferenciadas. A la izquierda se encuentra una lista con los diferentes tipos de gráficos, mientras que a la derecha se encuentra un texto que nos indica que seleccionemos una opción. Ambas partes se separan con una línea y para dotar de estilo se añade un GIF en la parte derecha.

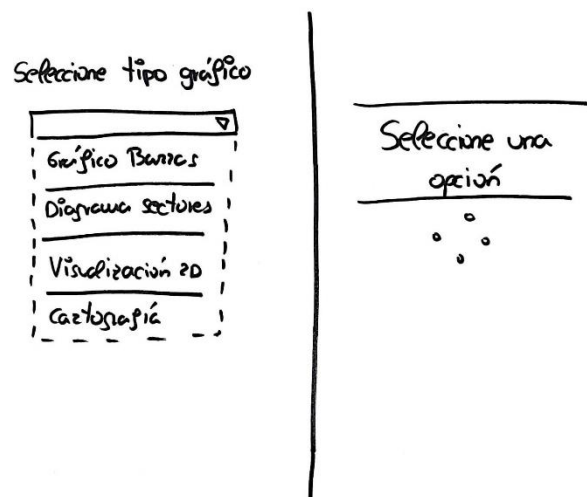


Ilustración 25: Boceto vista archivo.

En la Ilustración 26 se muestra la continuación del boceto anterior en el caso de que se seleccione un tipo de gráfico.



Ilustración 26: Boceto vista serie, selección de un gráfico.

Se puede observar cómo el lado de la derecha se queda en blanco, mientras que en el de la izquierda se muestra la información sobre el tipo de gráfico escogido. A continuación, hay dos entradas para seleccionar la frecuencia y los canales y, para acabar, la opción de guardar los datos con su correspondiente título, descripción y botón. Al final del todo se encuentra el botón de 'Mostrar': si el usuario ha introducido los parámetros correctamente y pulsa dicho botón, en una pantalla emergente aparecerá el gráfico en cuestión.

La 'vista\_archivo' es similar a la 'vista\_serie', recién comentada, con la diferencia de que está posee un botón de visualización de los datos (en vez de guardado de datos), el cual al pulsarlo muestra la información de forma tabulada en pantalla, y que el modo serie no dispone de cartografía cerebral.

Los bocetos de los gráficos se encuentran a continuación en la Ilustración 27, donde se observan los gráficos de dos canales, excepto en el caso de la cartografía cerebral, que se emplean todos los canales en el modo archivo.

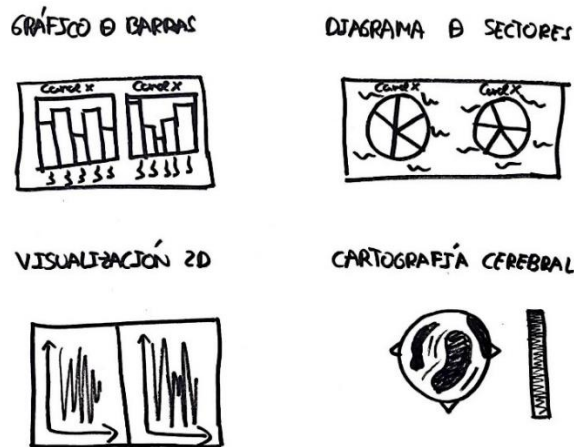


Ilustración 27: Bocetos de los diferentes tipos de gráficos.

Hay que mencionar que falta el boceto de la ‘vista\_error’, debido a su simplicidad, recordemos que esta vista surge por una incorrecta introducción de los datos por parte del usuario.

## 6.3 Implementación

A continuación, se detallan aspectos como el lanzamiento, el guardado de datos, los hilos y la representación gráfica en la aplicación.

### 6.3.1 Lanzamiento

La aplicación, como se ejecuta en local, se lanza mediante un fichero bat llamado launcher que se encuentra en la carpeta de la aplicación. Es una forma sencilla y rápida de ejecutarla. Al pulsar dicho archivo se generan dos hilos de ejecución, uno de ellos se encarga de cargar la aplicación en la ruta 127.0.0.1:5000, mientras que el otro hilo, que se mantiene dormido durante unos segundos, se encarga de abrir dicha ruta en el navegador Google Chrome. Esto se realiza simplemente para que, al ejecutar el fichero bat, la aplicación se abra con sus funcionalidades, en caso contrario habrá que recargar la página para que las adquiriera. Se ha observado que, tras encender el ordenador, la primera vez que se lanza dicho archivo, en ocasiones requiere más segundos de los implementados, por lo que hay que recargar la página.

Previo a la creación de los dos hilos mencionados, en la terminal de ejecución, tal y como podemos ver en Ilustración 28, se le pregunta al usuario si dispone de las librerías necesarias para poder emplear la aplicación. Este ha de responder con una ‘S’ de Si o una ‘N’ de No, en caso de que presione cualquier otro botón se asume que las posee y por lo tanto se puede lanzar la aplicación. Si el usuario comunica que no dispone las librerías, se le instalarán en esa misma terminal a continuación del texto mostrado.

Mencionar que, una vez que se instalan las bibliotecas en el ordenador no es necesario volver a instalarlas para cada ejecución de la aplicación, es decir, en teoría solo se deberán instalar en una ocasión, la primera vez que se use la aplicación. Si el usuario duda sobre si tiene instaladas las bibliotecas, puede pulsar ‘S’ y el sistema le comunicará cuales tiene ya instaladas e instalará el resto.

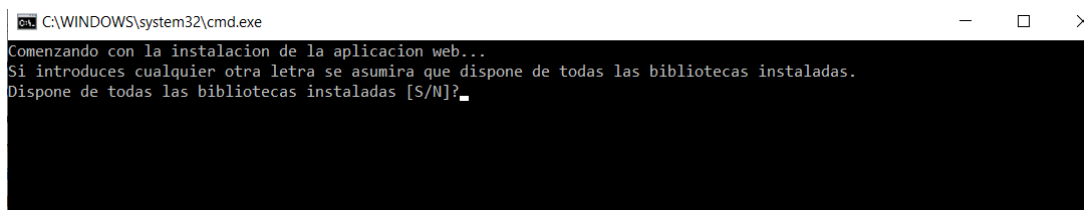


Ilustración 28: Terminal de lanzamiento.

### 6.3.2 Guardado de datos

La extracción de los datos, en primera instancia se realiza mediante el EEG, aunque la aplicación contiene datos de ejemplo por si el usuario inicialmente quiere probar la herramienta. En el modo ‘nuevo registro’ el usuario puede guardar nuevos datos, en el caso de que ya existían se sobrescribirán en el directorio ‘Datos’, dentro de la carpeta de la aplicación, almacenándose los 32 canales.

Hay que mencionar que tanto si el usuario ha elegido la opción ‘Guardado de datos’ como si ha visualizado un gráfico de dos canales o la cartografía, en el caso de guardarse los datos, éstos se almacenan en su forma original. Se implementa de esta forma para que, si el usuario elige posteriormente la opción ‘registro de archivo’, pueda visionar otros canales si lo precisa o realizar la cartografía cerebral, de esta forma se accede a un análisis más completo.

Al generar un nuevo registro, los datos guardados se sobrescriben, es decir, la aplicación únicamente mantiene los últimos datos tomados. Se guardan en ficheros de texto, uno por cada canal, con el siguiente formato: *FrecuenciaHz\_Canal\_nºcanal*, por ejemplo, *100Hz\_Canal\_12*. Si el usuario desea conservar los datos ha de copiarlos y asegurarlos en una carpeta o medio externo a la aplicación. En caso de querer usarlos posteriormente deberá remplazar los archivos de la carpeta ‘Datos’ por los que desee analizar con más detalle.

### 6.3.3 Hilos

Internamente, una vez que el usuario ha seleccionado el tipo de gráfico que quiere, se lanza un hilo con el archivo python en cuestión que genera una nueva pestaña con el gráfico. Esto se decide implementar así para que, en el caso de estar en ‘registro de archivo’, el usuario pueda abrir varios gráficos con diferente información y así poder hacer un análisis más completo, tal y como podemos ver en la Ilustración 29. Esto no se puede realizar en el modo serie ya que solo hay un EEG y el código está modularizado para realizar los gráficos de forma independiente (al inicio de cada uno se establece la comunicación con el EEG).



Ilustración 29: Ejemplo de la opcionalidad de abrir varios gráficos simultáneamente.

En lo que respecta al modo serie, se ha creado un hilo interno en los archivos para asegurar el correcto cierre de comunicación con el puerto serie ya que si no se cierra correctamente con su comando correspondiente, aparte de que no se cumplen los protocolos de comunicación con el EEG, no se puede empezar otra prueba en tiempo real, llegando al colapso de la máquina.

En la opción de guardado de datos del modo serie, para que el usuario decida cuando terminar el registro del EEG, se ha implementado una ventana con la biblioteca tkinter, que le permite

finalizar dicho proceso. Esta ventana esta provista con el texto ‘Click’ para finalizar el registro del EEG en tiempo real, tal y como podemos ver en la Ilustración 30.

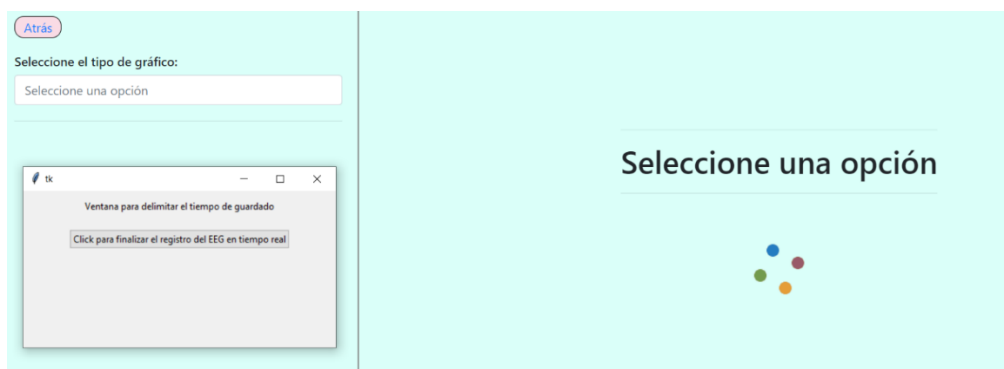


Ilustración 30: Ventana guardado de datos modo serie.

### 6.3.4 Representación gráfica

En el caso ‘nuevo registro’, a la hora de representar los gráficos, se toman del EEG un número de muestras representativas y se va actualizando el gráfico automáticamente en la misma ventana. En cambio, en el caso de ‘registro de archivo’ se implementa incluir un slider para que el usuario, mediante su desplazamiento, pueda ir actualizando los datos, tal y como se ven en la Ilustración 31.

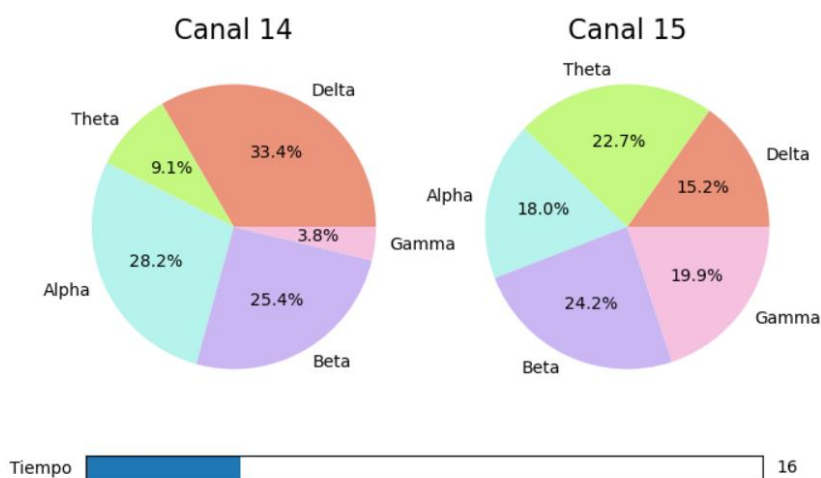


Ilustración 31: Visualización diagramas de sectores en modo archivo.

Cada paso del slider se corresponde con un nuevo conjunto de datos representativos, de forma que al moverlo un paso se actualiza el gráfico de ambos canales. Al final del slider se actualiza un número que se corresponde con el número de iteración, es decir el número de paso correspondiente.

Este slider se implementa como una barra deslizadora en todos los gráficos menos en ‘Visualización 2D’, donde se incluye dentro del propio gráfico, es decir, el usuario puede pulsar en cualquier parte de la representación y, desplazándose hacia la derecha o izquierda, mueve el gráfico en esa dirección.

A la hora de representar un gráfico desde el modo ‘registro de archivo’, en el caso de que un canal posea menos datos que otro de los representados, este mantendrá su valor mientras el resto se irán actualizando. En la Ilustración 32 encontramos un ejemplo de ello, el canal 11 mantiene su valor a partir del cuarto paso del slider mientras que el canal 10 cambia de valor en el siguiente.

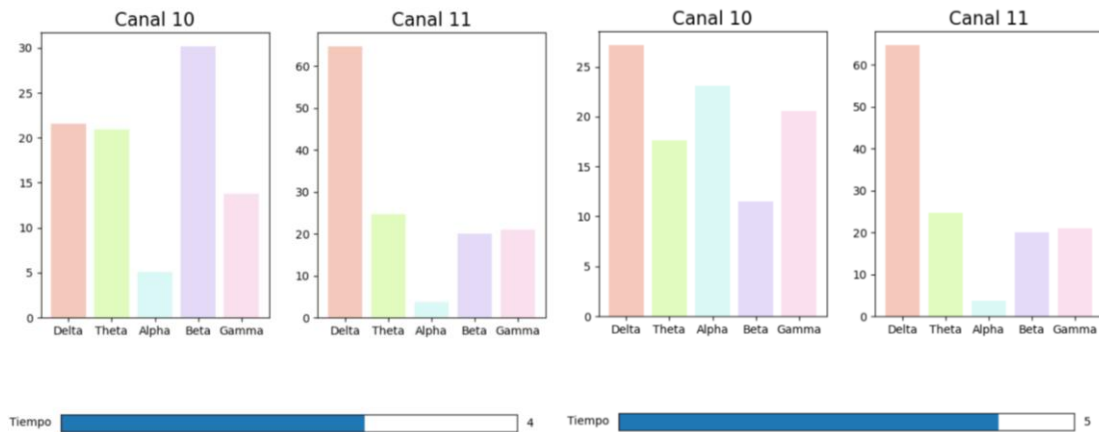


Ilustración 32: Visualización de gráficos de barras en modo archivo. (a) Paso cuatro del slider. (b) Paso 5 del slider.

Se ha comentado anteriormente que el gráfico de barras y el diagrama de sectores permiten que el usuario represente dos canales debido al uso de esta aplicación en el ámbito educativo, mientras que para representar la cartografía cerebral se emplean todos los canales, pero el caso de la visualización 2D es diferente, dependiendo del modo que seleccione el usuario. Si se toman datos del EEG, sobre todo en las frecuencias más altas, los datos se actualizan muy rápido no permitiendo el análisis de estos, además de la problemática de representar en tiempo. Por ello, en el caso de un nuevo registro se permite únicamente la visualización de dos canales, mientras que en el caso de cargar un registro de archivo se permiten todos los canales que desee el usuario para que pueda desplazarse por ellos para hacer un mejor análisis.

Para aportar robustez a la aplicación, en los casos que es necesario, si el usuario introduce más de dos canales, o el valor de los canales introducidos es superior a 31, se proporcionara una nueva vista al usuario, tal y como podemos ver en Ilustración 33, para avisarle de que se ha producido un error. El usuario tiene la posibilidad de regresar a la página anterior mediante el botón atrás y así poder rectificar los datos introducidos. Hay que mencionar que hay 32 canales, siendo el 0 el primero de ellos y el 31 el último.



Ilustración 33: Vista error.

La aplicación se ha implementado con una interfaz intuitiva y sencilla, pero para que sea útil, los datos han de ser interpretados por un especialista para poder sacar conclusiones de los diferentes gráficos. En el manual de usuario se encuentra una descripción detallada del resultado de la interfaz, lo que se corresponde con el Anexo 6. Además, se ha realizado la grabación de un vídeo que explica el funcionamiento de esta, el cual se encuentra en el [repositorio](#).



### 6.3.5 Atributos de calidad

Se examinan los atributos de calidad mencionados en [57], de los cuales se exponen a continuación los más destacados en la aplicación desarrollada:

- **Corrección:** es un atributo fundamental en cualquier aplicación, se basa en que el programa, al someterse a las tareas que requiera le usuario, las realice y ofrezca unos resultados correctos y fiables. Se ha verificado el correcto funcionamiento de los gráficos mostrados al usuario y, en el caso de emplear los mismos datos para representarlos, que exista repetitividad.
- **Usabilidad:** La aplicación es sencilla de entender y manejar, además gracias a los textos explicativos de ayuda incluidos en ella, el programa es muy intuitivo. Los menús y opciones creadas son sencillos de manejar teniendo la posibilidad de volver siempre a la ventana anterior.
- **Documentación:** Gracias al manual de usuario y el manual de instalación, existe una documentación completa de la aplicación que permite al usuario entender y resolver los problemas encontrados en el uso de la aplicación.
- **Comprensibilidad:** Este atributo se basa en que el software implementado sea entendible por otros desarrolladores. La aplicación posee una estructura lógica con un código legible y comprensible, además posee numerosos comentarios que permiten que se cumpla ese atributo.
- **Portabilidad:** El software implementado es portable a otros equipos, ya que incluye la descarga de bibliotecas en el caso de que el usuario no tenga instaladas las librerías necesarias para que la aplicación funcione.
- **Disponibilidad:** Hace referencia a si el sistema se ‘cae’ muy a menudo y el tiempo que emplea en arrancar. Los principales motivos por los que falla la aplicación son, en modo serie, cuando se rompe la comunicación con el EEG y, en modo archivo, cuando se intenta acceder al fichero de un canal y este no posee datos. Por lo que respecta al tiempo de arranque en cuestión de menos de diez segundos se puede arrancar o reiniciar el sistema.



## 7 Pruebas

En este apartado se especifican las distintas pruebas que se han realizado a lo largo del proyecto conforme este se desarrollaba, en las distintas fases de desarrollo. Incluye una prueba final, que se realiza una vez el código está completamente implementado.

El proyecto se ha desarrollado siguiendo los objetivos marcados de forma progresiva. A continuación, se explican las pruebas que se han realizado en las diferentes fases de desarrollo del proyecto, teniendo en la Tabla 6 un resumen de todas ellas.

- **Comunicación con el EEG y guardado de datos:** En primer lugar, se modificaron los ficheros de los drivers para que el ordenador reconociera el EEG: estos ficheros se llevaron a dos ordenadores y se verificó que ambos detectaban el EEG de forma correcta. La siguiente prueba se realizó mediante el programa SerialPlot, para conseguir la configuración del comando ‘Empezar una nueva prueba en tiempo real’ partiendo de otros comandos. Una vez adquirida la configuración, se implementó en lenguaje python y se verificó que se había establecido la comunicación. Se detectó un fallo cuando se recibían los datos y tras varias pruebas de análisis, se determinó que el problema estaba en el buffer, había que vaciarlo cada vez que se iniciara una nueva prueba porque si no se leían primero los datos acumulados en él. Inicialmente se pensaba que esto se hacía de forma automática, pero al no ser así, se modificó y se verificó el correcto funcionamiento. Inicialmente el puerto se introducía manualmente, otra prueba realizada fue implementar en el código la detección automática del puerto. Se probó cambiando el EEG de puerto y verificando que, pese a cambiar el número del puerto COM la comunicación se seguía estableciendo.

Posteriormente se establecieron las diferentes configuraciones para las distintas frecuencias y se implementaron en el código. Se verificó de forma manual que el número de datos por canal era el correcto.

El guardado inicial de los datos se realizaba en un solo fichero binario. Se realizaron diversas pruebas para determinar cuál era el mejor formato y extensión del archivo. Finalmente se implementó guardando los datos con una extensión txt para que fueran más transparentes al usuario, por lo que se realizó el cambio de bytes a enteros y se estableció un nuevo fichero para cada canal y frecuencia facilitando su entendimiento, ya que sino al guardar los datos en listas el formato era tedioso.

- **Gráficos de datos del EEG:** Se implementa el gráfico de visualización 2D directamente tomando datos del EEG y da diversos errores: para verificar el comportamiento correcto del gráfico se decide crear unos datos ejemplo para determinar que el comportamiento del gráfico es el adecuado. Una vez resueltas las incompatibilidades, se prueba directamente con el EEG para un canal y se obtienen resultados satisfactorios. Se hacen varias pruebas manuales de que los datos obtenidos y representados son los mismos.

La siguiente prueba se realiza modificando el código para poder representar varios canales en tiempo real. Cuando se hacen pruebas con pocos canales no ocurre nada destacable, pero cuando se aumenta el número de canales empiezan a ocurrir desfases entre los datos y pérdidas de información, llegando al punto en el que si abres otra aplicación el gráfico pierde los datos y mantiene el último valor encontrado. El problema está en que al final el ordenador pierde el ACK y por lo tanto no representa más datos. Para determinar cuántos canales se pueden representar a la vez se realizan pruebas aumentando el número de estos en tiempo real. El resultado es muy variable, pero se encuentra en el rango de dieciocho a veintidós canales representables sin incidencias de errores.

Para el gráfico de barras y diagrama de sectores se precisa hacer la FFT de los datos. Por ello, tras su implementación, se realiza una prueba introduciendo una sinusoidal formada por varias frecuencias como datos de entrada. Los datos obtenidos a la salida no

correspondían con los introducidos, pero gracias a diversas modificaciones del código, al introducir los datos creados se obtienen unos resultados acordes a ellos.

Se realiza una prueba en la que, para estos dos últimos tipos de gráficos comentados, se les introducen los mismos datos y se verifica que los resultados a la salida son los mismos, corroborando manualmente que corresponden con los datos ofrecidos por el programa.

- **Implementación en la aplicación:** Se realizan diversas pruebas para integrar el código anterior con todas sus funcionalidades en una aplicación. En primer lugar, se desarrollaron tests para verificar que el programa es capaz de captar los parámetros introducidos por el usuario en sus diferentes modalidades: radio buttons, entradas de texto, data-list...

Además, se comprueba, mediante la interacción de una persona ajena al proyecto, que los hilos se ejecutan de manera paralela e independiente.

En el modo archivo se implementa la actualización de los gráficos con una slider, realizándose una prueba para verificar que los datos de cada actualización corresponden con los del programa. Esto se ha realizado de forma manual.

Para el modo nuevo registro, a la hora de verificar que el puerto serie, tras una prueba se cerraba correctamente y se podía hacer consecutivamente otra, se encontraron problemas que llevaron a la modificación del código. Después de eso, se verifica que se pueden realizar y guardar pruebas consecutivas ya que se abre y cierra el puerto de forma correcta. Respecto al guardado de datos en tiempo real, se realizan varias pruebas implementando pop ups o entradas de texto: finalmente se determina que una ventana emergente era lo más visual e intuitivo para el usuario descartando que este tuviera que introducir el tiempo que debe durar el registro.

Fases de desarrollo	Pruebas
Comunicación con el EEG y guardado de datos	Modificación drivers
	Configuración de comando
	Puerto serie
	Comandos según frecuencias
	Guardado de datos
Gráficos de datos del EEG	Visualización 2D para un canal
	Formato de los bytes
	Visualización 2D para varios canales
	FFT de una sinusoidal
	Gráfico de barras y diagrama de sectores
Implementación en la aplicación	Captar datos del usuario
	Hilos
	Slider
	Cerrar puerto serie
	Guardado de datos en modo serie
Prueba final	Vídeo resumen

Tabla 6: Pruebas de las distintas fases de desarrollo.

## 7.1 Prueba final

Como prueba final se ejecuta la aplicación en un ordenador distinto, con su correspondiente usuario, que desconoce el funcionamiento del software implementado, de esta forma se verifica la usabilidad, la curva de aprendizaje y lo intuitiva que es la aplicación.

Previamente este usuario se ha leído el manual de instalación y el manual de usuario, que se encuentran en el *Anexo 5* y *Anexo 6*, entendiendo su comportamiento y demostrándolo en esa prueba. Hay que mencionar que el usuario en cuestión no poseía conocimientos de neurología por lo que el proceso no conllevó el entendimiento de los datos ni su análisis. Debido a la crisis

sanitaria existente, el software no ha podido ser probado por un especialista ni integrado en un aula sensorial.

Finalmente se realiza un video que documenta visualmente la aplicación, sus usos y funcionalidades, se encuentra disponible en el [repositorio](#) citado en el siguiente apartado.

## 8 Repositorio

En este apartado se describe la ubicación y archivos disponibles en el repositorio, necesarios para que la aplicación funcione correctamente y el usuario aprenda a utilizarla.

Dentro del repositorio encontramos:

- **Los drivers:** Se encuentran los dos archivos correspondientes a los drivers de instalación del EEG necesarios para que el ordenador lo reconozca. Han de instalarse los dos en el orden indicado en el manual de instalación.
- **El código de la aplicación:** Incluye el fichero bat que ejecuta la aplicación, los archivos HTML, los archivos de python y las carpetas donde se ubican los datos del EEG, inicialmente cargadas con datos para probar el modo archivo si así se requiere.
- **Vídeo:** Se encuentra un vídeo explicativo de la aplicación donde se mencionan sus utilidades, modos de funcionamiento, tipos de gráficos, etcétera.

En el repositorio se encuentra en: [REPOSITORIO TFM CLAUDIA GONZALO GIMENO](#)

## 9 Conclusiones

Al finalizar este proyecto, se puede decir que los objetivos planteados inicialmente han sido alcanzados, tanto desde el enfoque tecnológico como desde el ámbito médico-educativo.

Se ha realizado un estudio detallado sobre las distintas tecnologías de captación de información del funcionamiento cerebral, centrado finalmente en el EEG y el ámbito de la educación especial.

Se ha implementado correctamente la comunicación por puerto serie entre el EEG y el ordenador, con un conocimiento amplio sobre la configuración del comando 'Empezar una nueva prueba en tiempo real' y se ha entendido la distribución de datos en los diferentes canales y su forma de tramitarlos. Por lo tanto, los objetivos basados en la comunicación entre EEG y ordenador quedan alcanzados.

En lo que respecta a los datos, se ha determinado su estructura y su tratamiento para obtener información con significado médico. Se ha determinado que la potencia relativa es un parámetro que aporta claridad sobre el estado del sujeto y se han implementado una serie de gráficos que ayudan a visualizar los resultados aportando la claridad que necesitan los datos del EEG, ya que son muy complejos.

Por otro lado, se estima que la incidencia del TDAH en la población escolar abarca entre el 3 y 5%, además de que con una simple inspección visual a los datos obtenidos no se observa diferencias entre un paciente con o sin dicho trastorno [51], por lo que surge la necesidad de ofrecer una herramienta útil que permita al docente realizar su labor. La idea de esta aplicación es poder analizar el comportamiento de un niño en educación especial conforme se generan diversos estímulos sonoros, físicos o visuales, en tiempo real. De esta forma se puede determinar si su aprendizaje está siendo efectivo o si es necesario realizar modificaciones en el método o momento de enseñanza.

Lo más diferenciador de este trabajo es el empleo de únicamente dos electrodos en el lóbulo frontal, ya que permite disminuir enormemente el tiempo de preparación del niño, lo que facilita que aumente su frecuencia de uso. Hay que tener en cuenta que los niños con TDAH a menudo tienen pérdida de atención, impulsividad y son muy activos, por lo que, un análisis convencional de un EEG con todos sus canales, cables y preparación se puede ver interrumpido con facilidad. El implementar un software que este orientado a estos dos electrodos facilita el empleo de la herramienta EEG en el aula especial.

La aplicación que se ha desarrollado posee una interfaz clara y de fácil usabilidad para ayudar al experto a realizar su labor, permitiendo, dependiendo del modo de empleo, la visualización de varios gráficos (pudiendo seleccionar los canales), o representando los datos en tiempo real. El hecho de trabajar con dos canales proporciona una menor pérdida de la información. Hay que mencionar que, tener la posibilidad de analizar los gráficos posteriormente es una herramienta potente, ya que le permite al experto hacer un análisis más detallado de la información presentada. Por lo que se ha conseguido el objetivo relacionado con la solución software.

A nivel personal, este proyecto ha satisfecho mis expectativas, quedando pendiente su implantación y pruebas en un aula de educación especial que, debido a la COVID 19, no se ha podido realizar. Considero que este proyecto me ha ayudado a mejorar mi capacidad de programación, ya que nunca había realizado una aplicación, y me ha permitido repasar y aprender conceptos electrónicos, debido a que poseía una alta carga de hardware electrónico.

### 9.1 Futuras líneas de investigación

El objeto de este proyecto da por finalizado el alcance del trabajo. Como consideraciones futuras, se propone lo siguiente:

- Se plantea la preparación del software implementado con la herramienta EEG para su integración en una plataforma tecnológica de servicios para la educación especial.
- Se pretendía probar el software en un entorno real con niños en el ámbito de la educación especial, en concreto en un aula sensorial, pero, debido a la pandemia por la COVID 19, queda pospuesto este punto como trabajo futuro.
- En cooperación con Miguel Ángel Guillomía Gomez, doctorando de la Universidad de Zaragoza, crear una cartografía cerebral para los dos electrodos mencionados. Este punto también se vio paralizado por la pandemia.
- Estudiar si hay algún comando y su configuración que sea útil acorde al software implementado.
- Añadir más funcionalidades a la web, que pueda requerir el experto una vez la haya empleado.
- Tomar datos sobre sujetos con y sin TDAH, con el fin de investigar sobre este trastorno y como detectarlo.

# 10 Glosario

NIRS: espectroscopia de infrarrojo cercano (Near InfraRed Spectroscopy).

MEG: Magnetoencefalografía (Magnetoencephalography).

PET: Tomografía por emisión de positrones (Positron Emission Tomography).

EEG: Electroencefalograma (Electroencephalogram).

TDAH: Trastorno por déficit de atención e hiperactividad.

TEA: Trastorno del espectro autista.

BCI: Sistema de interfaz cerebro-computadora (Brain Computer Interfaces).

SNC: Sistema nervioso central.

UART: Transmisor-receptor asíncrono universal (Universal Asynchronous Receiver-Transmitter).

PID: Identificador de producto (Product ID).

ACK: Acuse de recibo (Acknowledgement).

FFT: Transformada rápida de Fourier (Fast Fourier Transform).

# 11 Bibliografía

- [1] «Serial Plot,» [En línea]. Available: <https://hackaday.io/project/5334-serialplot-realtime-plotting-software>. [Último acceso: 16 Mayo 2020].
- [2] «Anaconda web navigator,» [En línea]. Available: <https://www.anaconda.com/products/individual>. [Último acceso: 15 Junio 2020].
- [3] «Visual Studio Code,» [En línea]. Available: [https://www.ecured.cu/Visual\\_Studio\\_Code](https://www.ecured.cu/Visual_Studio_Code). [Último acceso: 20 Junio 2020].
- [4] «Python web,» [En línea]. Available: <https://desarrolloweb.com/articulos/1325.php>. [Último acceso: 25 Julio 2020].
- [5] «¿Qué es y para que sirve HTML?,» [En línea]. Available: <https://n9.cl/0hfc6>. [Último acceso: 29 Agosto 2020].
- [6] «¿Qué es CSS?,» [En línea]. Available: <https://www.hostinger.es/tutoriales/que-es-css/>. [Último acceso: 16 Septiembre 2020].
- [7] D. Kupas, B. Harangi, G. Czifra y G. Andrassy, «Decision support system for the diagnosis of neurological disorders based on gaze tracking,» de *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, Ljubljana, 2017.
- [8] M. I. Khalid, S. A. Aldosari y S. A. Alshebei, «Threshold based MEG data classification for healthy and epileptic subjects,» de *2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA)*, Ras Al Khaimah, 2016.

- [9] H. Sumiwaka, M. Izutsu y S. Hatakeyama, «Specific local sites in activity of the brain during listening to music,» de *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Seoul, 2012.
- [10] T. Yichao, D. Haishu, H. Lan y G. Qingcheng, «Monitoring Cerebral Oxygen Saturation Using Near Infrared Spectroscopy during Cardiopulmonary Bypass: Comparing with Monitoring Mixed Venous Oxygen Saturation,» de *2007 IEEE/ICME International Conference on Complex Medical Engineering*, Beijing, 2007.
- [11] X. Hou, H. Ding, Y. Teng, C. Zhou y D. Zhang, «NIRS study of cerebral oxygenation and hemodynamics in neonate at birth,» de *011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Boston, Boston, 2011.
- [12] W. Xiaohong, «A soft-sensing method for corn composition content using NIRS and LS-SVR,» de *2009 Chinese Control and Decision Conference*, Guilin, 2009.
- [13] T. Wojciechowski, G. Niedbała y M. Czechlowski, «Rapeseed seeds quality classification with usage of VIS-NIR fiber optic probe and artificial neural networks,» de *2016 International Conference on Optoelectronics and Image Processing (ICOIP)*, Warsaw, 2016.
- [14] D. Hwang, S. S. Nagarajan y O. Portnaguine, «Multiresolution MEG (magnetoencephalography) imaging by adaptive beamformer scanning,» de *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No.03CH37439)*, Cancun, 2003.
- [15] K. Chen, «Inter-Frame Co-Registration of Dynamically Acquired Fluoro-Deoxyglucose Positron Emission Tomography Human Brain Data,» de *2007 IEEE/ICME International Conference on Complex Medical Engineering*, Beijing, 2007.
- [16] G. H. Jin, G. R. Choi, . H. H. Park y . T. S. Lee, «Defining gross tumor volume using positron emission tomography/computed tomography phantom studies,» de *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)Osaka*, Osaka, 2013.
- [17] P. Buhler, U. Just, E. Will, J. Kotzerke y J. Van den Hoff, «An accurate method for correction of head movement in PET,» *IEEE Transactions on Medical Imaging*, vol. 23, n° 9, pp. 1176-1185, 2004.
- [18] Y. Picard y C. J. Thompson, «Motion correction of PET images using multiple acquisition frames,» *IEEE Transactions on Medical Imaging*, vol. 16, n° 2, pp. 137-144, 1997.
- [19] R. Fulton, «Accuracy of motion correction methods for PET brain imaging,» de *IEEE Symposium Conference Record Nuclear Science 2004*, 2004, 2004.
- [20] K. Thielemans, «Scatter estimation and motion correction in PET,» de *IEEE Nuclear Science Symposium Conference Record*, 2005, Fajardo, 2005.
- [21] C. Lin, Y. Wang, C. Fang, Y. Yu y J. King, «Extracting patterns of single-trial EEG using an adaptive learning algorithm,» de *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Milan, 2015.
- [22] S. G. Mangalagowri y P. C. P. Raj, «EEG feature extraction and classification using feed forward backpropagation algorithm for emotion detection,» de *2016 International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECOT)*, Mysuru, 2016.



- [23] Y. G. Chung, J. H. Kang y S. Kim, «Analysis of correlated EEG activity during motor imagery for brain-computer interfaces,» de *2011 11th International Conference on Control, Automation and Systems*, Gyeonggi-do, 2011.
- [24] M. Schneider, P. N. Mustaro y C. A. M. Lima, «Automatic recognition of epileptic seizure in EEG via support vector machine and dimension fractal,» de *International Joint Conference on Neural Networks*, Atlanta, 2009.
- [25] M. K. Hasan, R. Z. Rusho, T. M. Hossain y T. K. Gho, «Design and simulation of cost effective wireless EEG acquisition system for patient monitoring,» de *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*, Dhaka, 2014.
- [26] K. Wang, Y. Ho, Y. Huang y W. Fang, «Design of Intelligent EEG System for Human Emotion Recognition with Convolutional Neural Network,» de *2019 IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, Hsinchu, 2019.
- [27] F. Alba-Sanchez, O. Yanez-Suarez y H. Brust-Carm, «Assisted diagnosis of Attention-Deficit Hyperactivity Disorder through EEG bandpower clustering with self-organizing maps,» de *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, Buenos Aires, 2010.
- [28] H. Chen, J. Yan, Y. Gu, Y. Song y X. Li, «Mutual information analysis of EEG of children with attention-deficit/hyperactivity disorder,» de *2017 Chinese Automation Congress (CAC)*, Jinan, 2017.
- [29] L. Sekawati, N. U. Maulidevi y Suprijanto, «Machine learning to predict person's interest towards visual object by utilizing EEG signal,» de *2016 International Conference on Data and Software Engineering (ICoDSE)*, Denpasar, 2016.
- [30] Z. Song, B. Deng, J. Wang y R. Wang, «Biomarkers for Alzheimer's Disease Defined by a Novel Brain Functional Network Measure,» *IEEE Transactions on Biomedical Engineering*, vol. 66, n° 1, pp. 41-49, 2019.
- [31] C. Phang, C. Ting, S. B. Samdin y H. Ombao, «Classification of EEG-based Effective Brain Connectivity in Schizophrenia using Deep Neural Networks,» de *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, San Francisco, 2019.
- [32] H. Kim, «Mems Transducers for Non-Invasive Ultrasound Brain Stimulation,» de *2019 20th International Conference on Solid-State Sensors, Actuators and Microsystems & Eurosensors XXXIII (TRANSDUCERS & EUROSENSORS XXXIII)*, Berlin, 2019.
- [33] M. Honrath y A. Sabouni, «Study of intracranial pressure in human brain during transcranial magnetic stimulation,» de *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Milan, 2015.
- [34] G. Birot, L. Albera, D. Cosandier-Rimele y F. Wendl, «Localization of extended intracerebral current sources: Application to epilepsy,» de *009 IEEE International Workshop on Machine Learning for Signal Processing*, Grenoble, 2009.
- [35] G. Huiskamp, «EEG-MEG source characterization in post surgical epilepsy: the influence of large cerebrospinal fluid compartments,» de *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, San Francisco, 2004.
- [36] K. Sunthad, Y. Niitsu, M. Inoue y T. Yokemura, «Brain's Stress Observation System Using 2-Channels NIRS Based on Classroom Activity,» de *2019 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, 2019.

- [37] C. Li, M. Zhang, K. Ding, J. Zhou y D. Yu, «Effect of English learning experience on young children's prefrontal cortex functioning for attentional control: An fNIRS study,» de *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Berlin, 2019.
- [38] P. Hu y P. Kuo, «Adaptive learning system for E-learning based on EEG brain signals,» de *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*, Nagoya, 2017.
- [39] T. S. Petrenko, V. S. Kublanov y A. A. Petrenko and, «Application of the non-invasive adaptive neuroelectrostimulation device for treatment of cognitive impairment in the model of attention deficit hyperactivity disorder,» de *016 Cognitive Sciences, Genomics and Bioinformatics (CSGB)*, Novosibirsk, 2016.
- [40] C. Min, «Automatic detection and labeling of self-stimulatory behavioral patterns in children with Autism Spectrum Disorder,» de *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Seogwipo, 2017.
- [41] E. Abdulhay, M. Alafeef, H. Hadoush, N. Alomari y M. Bashayreh, «Frequency 3D mapping and inter-channel stability of EEG intrinsic function pulsation: Indicators towards autism spectrum diagnosis,» de *2017 10th Jordanian International Electrical and Electronics Engineering Conference (JIEEEEC)*, Amman, 2017.
- [42] W. K. Shams y A. Wahab, «Source-temporal-features for detection EEG behavior of autism spectrum disorder,» de *2013 5th International Conference on Information and Communication Technology for the Muslim World (ICT4M)*, Rabat, 2013.
- [43] P. Hoole , «Autism, EEG and brain electromagnetics research,» de *2012 IEEE-EMBS Conference on Biomedical Engineering and Sciences*, Langkawi, 2012.
- [44] A. Sahroni, T. Igasaki y N. Murayama, «Band powers analysis of spontaneous EEG with uncooperative autism children during short sleep condition,» de *2015 8th International Conference on Biomedical Engineering and Informatics (BMEI)*, Shenyang, 2015.
- [45] J. Fan, «A Step towards EEG-based brain computer interface for autism intervention,» de *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Milan, 2015.
- [46] R. Gabriel, M. Machado Spindola, A. Mesquita y A. Zerbetto Neto, «Identification of ADHD Cognitive Pattern Disturbances Using EEG and Wavelets Analysis,» de *2017 IEEE 17th International Conference on Bioinformatics and Bioengineering (BIBE)*, Washington, 2017.
- [47] S. Rodrak y Y. Wongsawat, «EEG brain mapping and brain connectivity index for subtypes classification of attention deficit hyperactivity disorder children during the eye-opened period,» de *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Osaka, 2013.
- [48] A. Alex, S. Coelli, A. M. Bianchi, L. Ponzini y M. P. Canevini, «EEG analysis of brain activity in attention deficit hyperactivity disorder during an attention task,» de *2017 IEEE 3rd International Forum on Research and Technologies for Society and Industry (RTSI)*, Modena, 2017.
- [49] S. Khoshnoud, M. Shamsi y M. A. Nazari, «Non-linear EEG analysis in children with attention-deficit/ hyperactivity disorder during the rest condition,» de *2015 22nd Iranian Conference on Biomedical Engineering (ICBME)*, Tehran, 2015.

- [50] P. Hu, P. Chen y P. Kuo, «Educational Model Based on Hands-on Brain-Computer Interface: Implementation of Music Composition Using EEG,» de *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Miyazaki, Japan, 2018.
- [51] M. Navascués, M. Sebastián y J. Valdizán, «Fractal and Smooth Complexities in Electroencephalographic Processing,» *OMICS Publishing Group*; <http://dx.doi.org/10.4172/2168-9679.1000198>, vol. 4, 2015.
- [52] «Pyqtgraph,» [En línea]. Available: <http://www.pyqtgraph.org/>. [Último acceso: 15 Agosto 2020].
- [53] J. Cabanyes-Truffino, «Cartografía cerebral: metodología y aplicaciones en la clínica neurológica J. Cabanyes-Truffino,» *REV NEUROL*, vol. 28, nº 11, pp. 1090-1098, 199.
- [54] C. Sheikholeslami,, H. Yuan, E. He, X. Bai, L. Yang y B. He, «A High Resolution EEG Study of Dynamic Brain Activity,» de *IEEE EMBS*, Lyon, France, 2007.
- [55] J. Valdizán, M. Navascués y M. Sebastián, «Distribución espectral cortical del electroencefalograma infantil,» *REV NEUROL*, vol. 28, nº 4, pp. 361-365, 1999.
- [56] «Modelo vista controlador,» [En línea]. Available: <https://desarrolloweb.com/articulos/que-es-mvc.html>. [Último acceso: 30 Agosto 2020].
- [57] «Atributos de calidad de una aplicación web,» [En línea]. Available: <https://ddw.com.ar/blog/113-tecnologia-software-aplicaciones-y-servicios-web/423-test-de-calidad-de-aplicaciones-web>. [Último acceso: 12 Septiembre 2020].

## 12 Índice de tablas

Tabla 1: Cronograma.....	2
Tabla 2: Comandos EEG.....	16
Tabla 3: Configuración del comando 'Empezar un aprueba en tiempo real' .....	16
Tabla 4: Comando 'Empezar una prueba en tiempo real', 500 Hz, 32 canales.....	17
Tabla 5: Características puerto serie .....	20
Tabla 6: Pruebas de las distintas fases de desarrollo.....	38

## 13 Índice de ilustraciones

Ilustración 1: Sistema de colocación de electrodos diez y veinte. (a) Vista superior. (b) Vista lateral. [9] .....	4
Ilustración 2: Configuración del conjunto de sensores SQUID [14].....	5
Ilustración 3:Sistema de seguimiento del paciente en escáner PET. (a) Vista de frente. (b) Vista del paciente. [16].....	5
Ilustración 4: Sistema internacional 10-20 de colocación de electrodos en EEG. [24].....	6
Ilustración 5: Estructura general de la neurona y generación de señal eléctrica debido a estimulación externa [25].....	6
Ilustración 6: Sistema de medición EEG. (a) Características del sistema, vista de frente. (b) Globo de presión, vista superior. [21] .....	7
Ilustración 7: Sensor NIRS HOT-1000. [36].....	8
Ilustración 8: Ejemplo mapa topológico. ....	9
Ilustración 9: EEG Bitmed eXea Ultra.....	11
Ilustración 10: Electrodos EEG. (a) Gorro de electrodos. (b) Conexión gorro al EEG. ....	11
Ilustración 11: Leds indicadores. ....	12
Ilustración 12: Led indicador de batería.....	12
Ilustración 13: Entradas EEG.....	12
Ilustración 14: Cables y conexiones del EEG. ....	13
Ilustración 15: Chip FT232BL. (a) Imagen chip FT232BL. (b) Esquemático chip FT232BL ..	14
Ilustración 16: Captura de la aplicación SerialPlot, se corresponde con los primeros datos recibidos del EEG. ....	15
Ilustración 17:Visualización 2D de un canal, 500Hz. ....	21
Ilustración 18:Visualización 2D de todos los canales, 500Hz. ....	22
Ilustración 19: Ejemplo de la distribución de los datos sobre las bandas de frecuencia. ....	23
Ilustración 20: Gráficos FFT. (a) Gráfico de barras. (b) Diagrama de sectores. ....	24
Ilustración 21: Ejemplo cartografía cerebral. ....	24
Ilustración 22: Diagrama de flujo de lanzamiento de la aplicación. ....	28
Ilustración 23: Diagrama de flujo de la aplicación.....	29
Ilustración 24: Boceto vista inicio.....	30
Ilustración 25: Boceto vista archivo.....	31
Ilustración 26: Boceto vista serie, selección de un gráfico. ....	31
Ilustración 27: Bocetos de los diferentes tipos de gráficos. ....	32
Ilustración 28: Terminal de lanzamiento.....	32
Ilustración 29: Ejemplo de la opcionalidad de abrir varios gráficos simultáneamente. ....	33
Ilustración 30: Ventana guardado de datos modo serie.....	34
Ilustración 31: Visualiación diagramas de sectores en modo archivo.....	34
Ilustración 32: Visualización de gráficos de barras en modo archivo. (a) Paso cuatro del slider. (b) Paso 5 del slider.....	35
Ilustración 33: Vista error. ....	35

## 14 Índice de ecuaciones

Ecuación I.....	17
Ecuación II .....	17
Ecuación III.....	17
Ecuación IV.....	18
Ecuación V .....	18
Ecuación VI.....	18
Ecuación VII .....	18
Ecuación VIII.....	18
Ecuación IX.....	18
Ecuación X.....	18
Ecuación XI.....	19
Ecuación XII .....	19

## 15 Anexos

1. Bibliotecas y paquetes.
2. Configuración comando ‘Empezar una prueba en tiempo real’ con diferentes frecuencias
3. Prueba FFT.
4. Diagramas de flujo.
5. Manual de instalación.
6. Manual de usuario.
7. Datasheet manual EEG.