



# Firmware de dispositivo de adquisición de señales neurofisiológicas

Autor:

Leandro Arrieta

Director:

Diego Coulombie (UNLaM)

Codirector:

Ariel Gentile (Advantek)

*Esta planificación fue realizada en el curso de Gestión de proyectos  
entre el 24 de junio de 2021 y el 19 de agosto de 2021.*

## Índice

1. Descripción técnica-conceptual del proyecto a realizar . . . . .	5
2. Identificación y análisis de los interesados . . . . .	6
3. Propósito del proyecto . . . . .	6
4. Alcance del proyecto . . . . .	6
5. Supuestos del proyecto. . . . .	7
6. Requerimientos . . . . .	7
7. Historias de usuarios ( <i>Product backlog</i> ). . . . .	8
8. Entregables principales del proyecto . . . . .	10
9. Desglose del trabajo en tareas . . . . .	10
10. Diagrama de Activity On Node. . . . .	12
11. Diagrama de Gantt . . . . .	12
12. Presupuesto detallado del proyecto . . . . .	14
13. Gestión de riesgos . . . . .	14
14. Gestión de la calidad . . . . .	16
15. Procesos de cierre . . . . .	19

## Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	01/07/2021
1.0	Se completa hasta la sección 5 inclusive	06/07/2021
2.0	Se agregan supuestos en la sección 5 Se completa hasta la sección 9 inclusive	13/07/2021
3.0	Se agregan requerimientos de documentación Se completa hasta el punto 12 inclusive	26/07/2021
4.0	Se agrega un ítem en la sección 4 Se agrega un ítem en la sección 5 Se completa el plan	02/08/2021

## Acta de constitución del proyecto

Buenos Aires, 24 de junio de 2021

Por medio de la presente se acuerda con el Ing. Leandro Arrieta que su Trabajo Final de la Carrera de Especialización en Sistemas Embebidos se titulará “Firmware de dispositivo de adquisición de señales neurofisiológicas”, consistirá esencialmente en el desarrollo y la implementación del firmware del dispositivo perteneciente al proyecto de investigación “C2-ING-066 Herramientas de uso comunitario para el desarrollo de la industria de Tecnología Neurofisiológica”, y tendrá un presupuesto preliminar estimado de 630 hs de trabajo, con fecha de inicio 24 de junio de 2021 y fecha de presentación pública 15 de mayo de 2022.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg  
Director posgrado FIUBA

Diego Coulombie  
UNLaM

## 1. Descripción técnica-conceptual del proyecto a realizar

La Neurofisiología es una rama de las neurociencias, que se encarga del estudio funcional de la actividad bioeléctrica del sistema nervioso central, periférico y autonómico, mediante la utilización de equipos y técnicas de análisis avanzado, como la Electroencefalografía (EEG), la Electromiografía (EMG), los Potenciales Evocados (PE), la Polisomnografía (PSG) y otras nuevas técnicas como el neuromonitoreo (NM) o la medición de profundidad de anestesia (MPA). En el país hay al menos 4 empresas que diseñan y fabrican este equipamiento no existiendo un caso similar de competencia en otros países de la región. El costo de desarrollar un equipamiento médico siempre fue elevado. Hace ya muchos años que se implementan regulaciones a los productos que son cada vez mas exigentes en materia de seguridad y eficacia. Sumado a esto los cambios tecnológicos en electrónica y comunicaciones, causan que las empresas locales no puedan seguir esa evolución por ser proyectos económicamente inviables, dejando que sus productos con diseños obsoletos sean paulatinamente expulsados del mercado por su pobre demanda.

El proyecto de investigación "C2-ING-066 Herramientas de uso comunitario para el desarrollo de la industria de Tecnología Neurofisiológica" de la UNLaM propone generar una plataforma de adquisición de señales neurofisiológicas, que sea de uso común para todos los fabricantes de equipos del subsector, para investigación en las universidades y para el eventual desarrollo de nuevos productos y nuevas empresas tecnológicas. El alcance de la plataforma facilitaría el cumplimiento de los requisitos regulatorios, como los de seguridad, análisis de riesgos y compatibilidad electromagnética, dejando a cargo del fabricante la adecuación de ergonomía y usabilidad para cada uso previsto. La existencia de un dispositivo de uso común cuyo costo de diseño, manufactura y ensayos se amortiza entre varios fabricantes, da la posibilidad de destinar mas recursos en actividades que generen un mayor valor agregado, como software más complejo, innovación en algoritmos o interfaces más eficientes y seguras. De esta manera se potencia al subsector industrial en particular y al de las neurociencias en general. En el marco de ese proyecto nace esta propuesta para darle vida al primer prototipo que se tiene armado pero que le falta la programación del sistema embebido.

El dispositivo de adquisición de señales neurofisiológicas a desarrollar, a partir de ahora llamado DASN, será parte de un sistema médico, por lo que el firmware se debe desarrollar cumpliendo los estándares de tal industria. En la figura 1 se puede ver un diagrama en bloques del sistema. Se observa que el sistema estará formado por un equipo de registro, por el DASN y eventualmente por un estimulador externo.

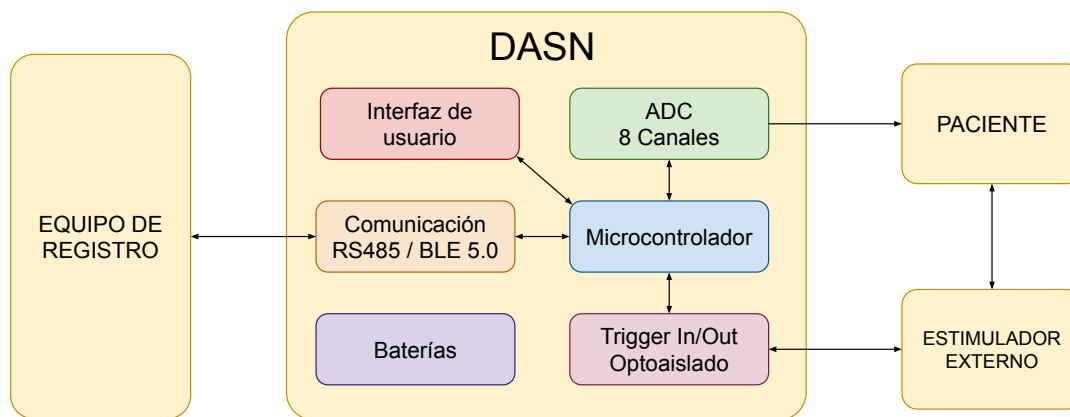


Figura 1. Diagrama en bloques del sistema.

El DASN tendrá 8 canales de entrada para adquirir las señales neurofisiológicas del paciente y las transmitirá al equipo de registro de manera inalámbrica a través de una comunicación BLE 5.0, o cableada a través de una comunicación RS485. Las señales adquiridas se podrán sincronizar con un estimulador externo, pudiendo funcionar el estimulador como master o slave. La interfaz de usuario solo dará indicaciones de encendido, apagado y estado de funcionamiento, ya que las señales adquiridas se visualizarán en el monitor del equipo de registro. El DASN se alimentará a baterías.

## 2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Diego Coulombie	UNLaM	Director del proyecto
Responsable	Leandro Arrieta	FIUBA	Alumno
Orientador	Ariel Gentile	Advantek	Codirector trabajo final

- Dr. Ing. Diego Coulombie: Tiene experiencia en la redacción de documentos académicos. Puede ayudar en la elaboración de la memoria descriptiva final del proyecto.
- Ing. Ariel Gentile: Tiene experiencia en programación. Puede ayudar en el desarrollo de los módulos de software.

## 3. Propósito del proyecto

El propósito de este proyecto es poner en práctica todos los conocimientos adquiridos durante la carrera de especialización, colaborar con el proyecto "C2-ING-066 Herramientas de uso comunitario para el desarrollo de la industria de Tecnología Neurofisiológica", y a su vez, completar los requisitos de aprobación de la CESE.

## 4. Alcance del proyecto

El presente proyecto incluye:

- Diseño del firmware embebido del DASN.
- Diseño del protocolo de comunicación entre DASN y equipo de registro.
- Documentación acorde a norma ISO 62304.
- Software de prueba para simular un equipo de registro y probar el DASN.

El presente proyecto no incluye:

- Diseño de hardware.
- Filtrado digital de las señales adquiridas.
- Validación de norma ISO 62304.

## 5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- El cliente proveerá del hardware necesario para el desarrollo del firmware.
- El protocolo de comunicación entre el DASN y el equipo de registro es de libre diseño.
- El DASN y el equipo de registro estarán siempre cerca por lo que no hay requisitos de alcance en la comunicación inalámbrica.
- El hardware no cambiará durante el desarrollo del proyecto.
- La validación de la norma ISO 62304 en laboratorio acreditado queda a cargo del cliente.

## 6. Requerimientos

### 1. Requerimientos de interfaces externas

- 1.1. El software deberá comunicarse con el equipo de registro utilizando una interfaz BLE 5.0.
- 1.2. El software deberá comunicarse con el equipo de registro utilizando una interfaz RS485.
- 1.3. El software deberá indicar mediante el led1 que está transmitiendo las señales adquiridas de forma inalámbrica o cableada.
- 1.4. El software deberá indicar mediante el led2 si está encendido o apagado. También deberá indicar con el mismo led2 si entra en el modo pairing BLE.
- 1.5. El software deberá manejar una salida para sincronizar la adquisición con un estimulador externo (dispositivo externo como slave). La salida deberá poder configurarse entre normal bajo y normal alto. El pulso deberá poder configurarse entre 5 anchos de pulso diferentes (0,1 ms; 0,5 ms; 1 ms; 5 ms; 10 ms).
- 1.6. El software deberá manejar una entrada para sincronizar la adquisición con un estimulador externo (dispositivo externo como master). La detección deberá ser por flanco y se deberá poder configurar si el flanco es de subida o de bajada.
- 1.7. El software deberá manejar el pulsador que servirá para encender el dispositivo y para realizar el pairing BLE.
- 1.8. El software deberá generar la señal de impedancia para los canales de potenciales evocados.
- 1.9. El software deberá medir el estado de las baterías con el ADC interno del MCU.

### 2. Requerimientos funcionales

- 2.1. El software deberá configurar la frecuencia de muestreo de la señal adquirida entre 7 diferentes valores ( el ADC de "EEG-FrontEnd\_v1.x.0.SchDoc" ofrece 65 Hz, 131 Hz, 262 Hz, 524 Hz, 1048 Hz, 2096 Hz y 4193 Hz).
- 2.2. El software deberá poder adquirir de 1 a 8 canales simultáneos.
- 2.3. Mediante comandos recibidos por BLE 5.0 o RS485 el software deberá poder iniciar y parar la adquisición.

- 2.4. El software deberá configurar la ganancia de amplificación de cada canal entre 7 diferentes valores ( el ADC de "EEG-FrontEnd\_v1.x.0.SchDoc" ofrece 1, 2, 4, 6, 8, 12 y 24).
- 2.5. El software deberá configurar el ADC para usar una tipología de entrada diferencial o referencial.
- 2.6. El software deberá medir la impedancia de los electrodos con una señal de medición de impedancia cuya frecuencia deberá ser de 7,8 Hz o 31,2 Hz.
- 2.7. El software deberá seleccionar a qué electrodos le inyecta la señal de medición de impedancia.
- 2.8. El software deberá seleccionar para cada electrodo si usa la señal de impedancia generada por el ADC o la generada por el MCU.
- 2.9. El software deberá enviar la siguiente información de autodiagnóstico: temperatura del ADC, valor de las tensiones del ADC y frecuencia del clock del ADC.
- 2.10. El software deberá prender y apagar el equipo con una pulsación corta del botón, pulsación menor a 1 segundo.
- 2.11. El software deberá entrar en el modo de apareo de la comunicación BLE con una pulsación larga del botón, pulsación mayor a 4 segundos.
- 2.12. Luego de 1 minuto de inactividad, el software deberá entrar a un modo de bajo consumo de energía. Para esto deberá apagar el ADC, el transceiver RS485 y la alimentación de todos los periféricos externos al MCU.
- 2.13. El software deberá manejar una comunicación RS485 hasta 3 MBd.
- 2.14. El software deberá poder recibir comandos mientras está transmitiendo las señales adquiridas.

### 3. Requerimientos de documentación

- 3.1. La documentación debe cumplir los requisitos de la norma ISO 62304.
- 3.2. Toda la documentación del proyecto se almacenará bajo un sistema de control de versiones GIT.
- 3.3. Toda la documentación del código se realizará utilizando la herramienta Doxygen.
- 3.4. Se deberá realizar un informe de avance del proyecto en el séptimo mes de trabajo
- 3.5. Se deberá realizar la memoria técnica del trabajo final con la plantilla elaborada por la cátedra de gestión de proyecto.

## 7. Historias de usuarios (*Product backlog*)

Criterio para ponderar los *story points*:

Se deben evaluar los siguientes aspectos de cada historia de usuario con una escala de 1 a 3, donde 1 es la calificación más baja y 3 la más alta:

- **Dificultad** (cantidad de trabajo a realizar).
- **Complejidad** (nivel de sofisticación del trabajo).
- **Incertidumbre** (nivel de riesgo que involucra realizar la tarea).



Finalmente se determinan los *story points* realizando la siguiente operación:

$$\text{Story points} = \text{Dificultad} \times \text{Complejidad} \times \text{Incertidumbre}$$

1. "Como usuario quiero seleccionar la frecuencia de muestreo y ganancia de cada canal para poder hacer distintos tipos de estudios neurológicos."

- **Dificultad:** 1
- **Complejidad:** 2
- **Incertidumbre:** 1

$$\text{Story points} = 2$$

2. "Como usuario quiero que el equipo sea inalámbrico para que el paciente se mueva libremente"

- **Dificultad:** 2
- **Complejidad:** 3
- **Incertidumbre:** 3

$$\text{Story points} = 18$$

3. "Como usuario quiero medir la impedancia de los electrodos para poder comprobar su correcta colocación."

- **Dificultad:** 2
- **Complejidad:** 2
- **Incertidumbre:** 1

$$\text{Story points} = 4$$

4. "Como usuario quiero sincronizar el equipo con un estimulador externo para hacer estudios de potenciales evocados."

- **Dificultad:** 2
- **Complejidad:** 2
- **Incertidumbre:** 2

$$\text{Story points} = 8$$

5. "Como usuario quiero ver el estado de las baterías para saber la autonomía del equipo."

- **Dificultad:** 2
- **Complejidad:** 2
- **Incertidumbre:** 2

$$\text{Story points} = 8$$

## 8. Entregables principales del proyecto

Los entregables del proyecto son:

- Código fuente del firmware.
- Código fuente del Software de prueba.
- Documentación acorde a ISO 62304.
- Informe final.

## 9. Desglose del trabajo en tareas

En cada grupo de tareas se indica el subtotal de horas correspondiente al mismo.

### 1. Planificación y documentación (100 hs)

- 1.1. Planificación del proyecto (20 hs)
- 1.2. Especificación de requisitos de software (20 hs)
- 1.3. Investigación Norma ISO 62304 (30 hs)
- 1.4. Análisis de riesgo (30 hs)

### 2. Diseño e implementación del firmware (270 hs)

- 2.1. Preparación del entorno de desarrollo (20 hs)
- 2.2. Diseño de la arquitectura del firmware (30 hs)
- 2.3. Investigación sobre tecnología BLE (20 hs)
- 2.4. Diseño del protocolo de comunicación (30 hs)
- 2.5. Desarrollo del módulo comunicación BLE (40 hs)
- 2.6. Desarrollo del módulo comunicación RS485 (20 hs)
- 2.7. Desarrollo del módulo ADC (30 hs)
- 2.8. Desarrollo del módulo manejo estimulador externo (20 hs)
- 2.9. Desarrollo del módulo interfaz de usuario (10 hs)
- 2.10. Pruebas de integración (30 hs)
- 2.11. Pruebas de verificación (20 hs)

### 3. Diseño e implementación software de prueba (150 hs)

- 3.1. Preparación del entorno de desarrollo (20 hs)
- 3.2. Diseño de la arquitectura (20 hs)
- 3.3. Desarrollo de drivers (40 hs)
- 3.4. Desarrollo de Interfaz gráfica (30 hs)
- 3.5. Pruebas de integración (20 hs)
- 3.6. Pruebas de verificación (20 hs)

**4. Cierre del proyecto (110 hs)**

- 4.1. Pruebas de validación (40 hs)
- 4.2. Confección de informes de avance del proyecto (10 hs)
- 4.3. Presentaciones al cliente (10 hs)
- 4.4. Elaboración de la memoria descriptiva final del proyecto (40 hs)
- 4.5. Presentación final del proyecto (10 hs)

**Cantidad total de horas: (630 hs)**

## 10. Diagrama de Activity On Node

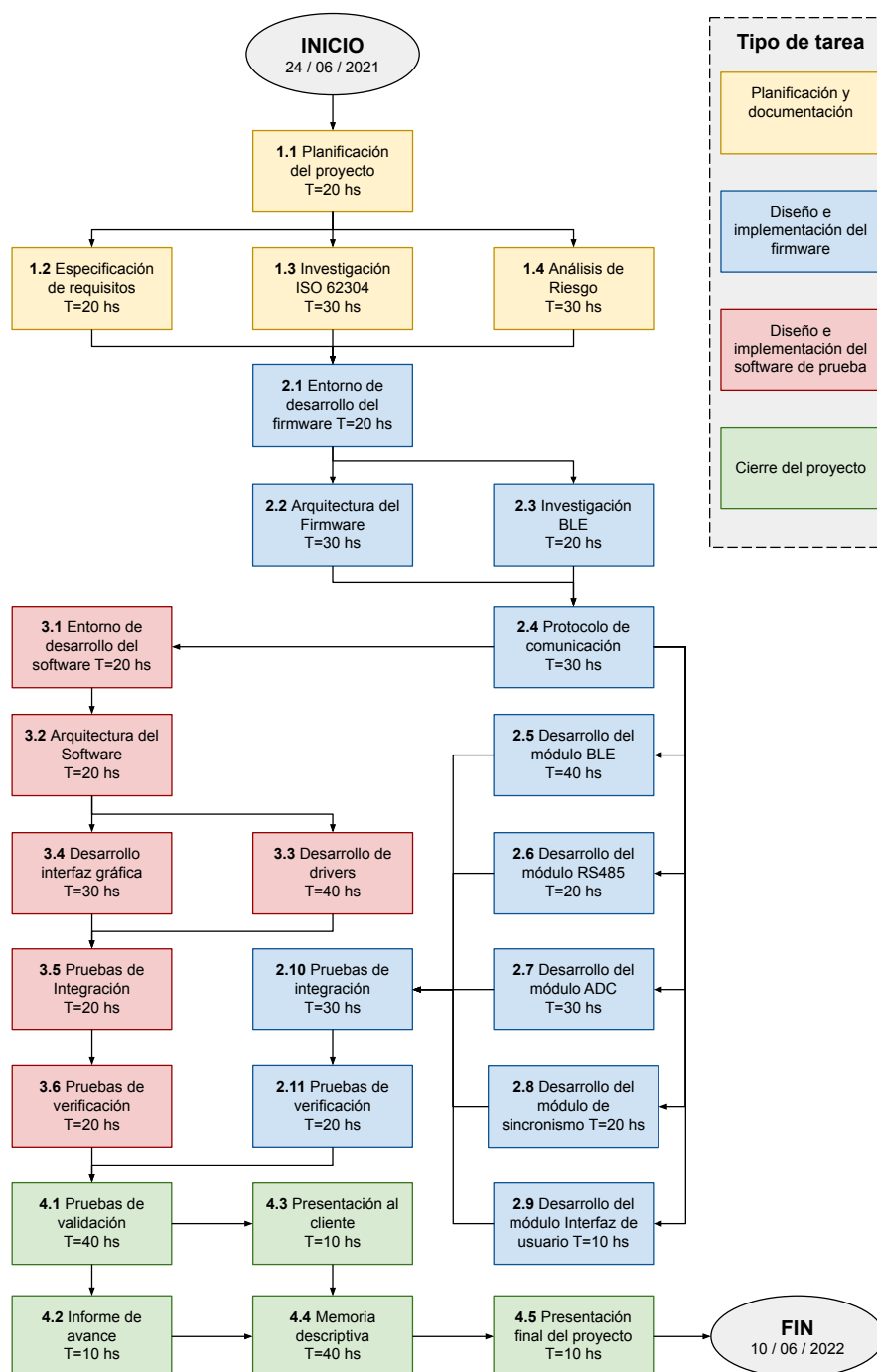


Figura 2. Diagrama de *Activity on Node*.

## 11. Diagrama de Gantt

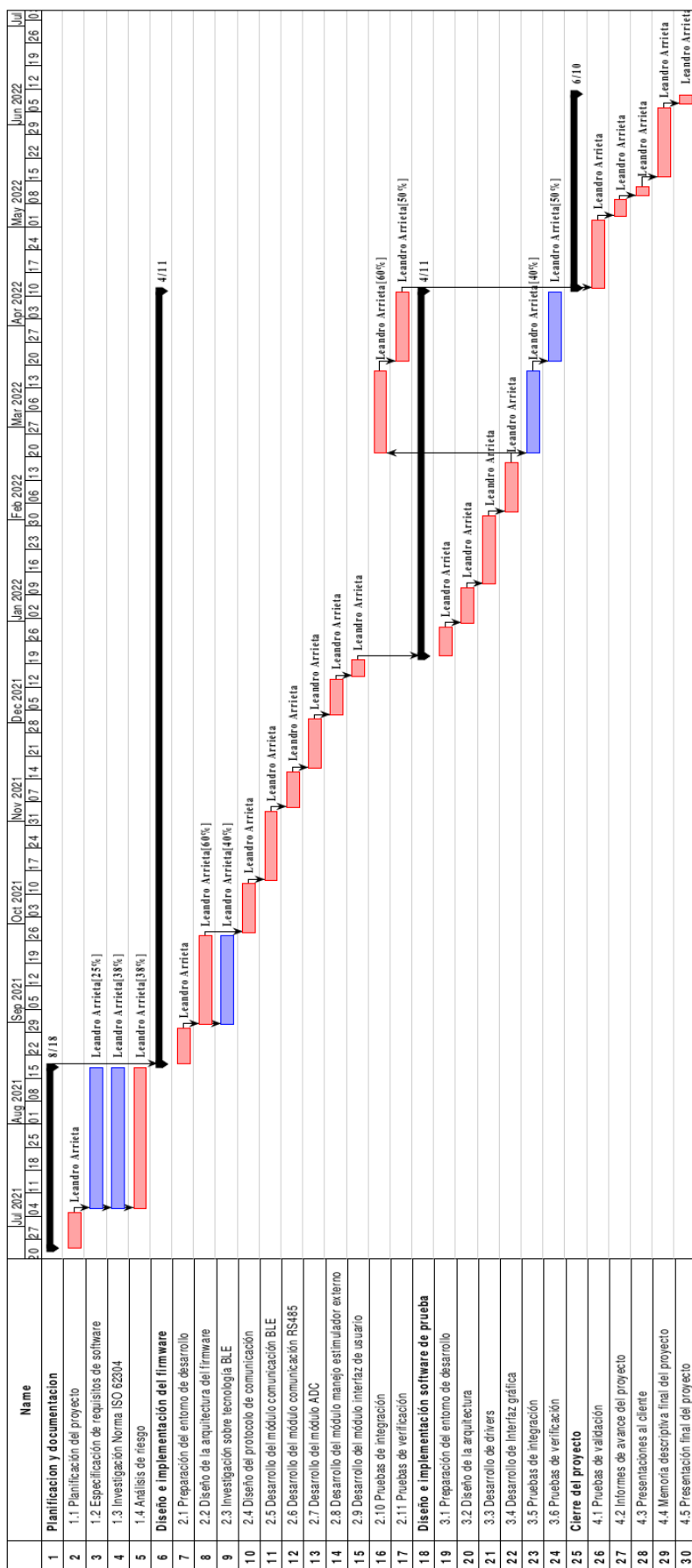


Figura 3. Diagrama de Gantt

## 12. Presupuesto detallado del proyecto

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Mano de obra	630	u\$s 10,00	u\$s 6300,00
Módulo de desarrollo LAUNCHXL-CC2640R2	2	u\$s 34,80	u\$s 69,60
Módulo de desarrollo ADS1299EEGFE-PDK	1	u\$s 238,80	u\$s 238,80
SUBTOTAL			u\$s 6608,40
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Costos indirectos, estimados como 20 % de los los costos directos	1	u\$s 1321,68	u\$s 1321,68
SUBTOTAL			u\$s 1321,68
TOTAL			u\$s 7930,08

## 13. Gestión de riesgos

Criterio para ponderar los riesgos:

- Severidad (S): Rango de 1 a 10. Mientras más severo, más alto es el número.
- Probabilidad de ocurrencia (O): Rango de 1 a 10. Mientras más probable, más alto es el número.

a) A continuación se listan los riesgos asociados al proyecto:

**Riesgo 1:** Retraso en la fecha de finalización del proyecto.

- **Severidad (10):** Implica incumplimiento con el cliente y la no aprobación de la CESE.
- **Ocurrencia (5):** El responsable del proyecto no posee experiencia en la programación de sistemas embebidos por lo que la ejecución de las tareas puede llevar más tiempo del planificado.

**Riesgo 2:** Errores en el diseño del hardware impiden probar el firmware.

- **Severidad (7):** La falta de prototipo demora las pruebas de validación.
- **Ocurrencia (7):** Quienes desarrollaron el hardware no tenían experiencia en el desarrollo de equipos con el MCU seleccionado ni con equipos inalámbricos.

**Riesgo 3:** Daños en el prototipo de hardware durante el desarrollo.

- **Severidad (7):** La falta de prototipo demora las pruebas de validación.
- **Ocurrencia (2):** Existen 3 prototipos armados por lo que la probabilidad de quedarse sin prototipo para las pruebas es baja.

**Riesgo 4:** No cumplir con todos los requerimientos.

- **Severidad (9):** El producto final no funcionará como se espera.

- **Ocurrencia (5):** El responsable del proyecto no posee experiencia en la programación de sistemas embebidos por lo que la ejecución de las tareas puede llevar más tiempo del planificado.

**Riesgo 5:** El rendimiento del hardware seleccionado no es suficiente para cumplir con todos los requisitos.

- **Severidad (9):** El producto final no funcionará como se espera.
- **Ocurrencia (2):** Luego de analizar las hojas de datos de los principales componentes y ver los ejemplos de aplicación del fabricante se infiere una baja probabilidad de ocurrencia de este riesgo.

b) Tabla de gestión de riesgos: (El RPN se calcula como  $RPN = S \times O$ )

Riesgo	S	O	RPN	S*	O*	RPN*
1	10	5	50	10	3	30
2	7	7	49	5	5	25
3	7	2	14			
4	9	5	45	9	3	27
5	9	2	18			

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 30.

Nota: los valores marcados con (\*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

**Riesgo 1:**

**Plan de mitigación:** Se efectuará un seguimiento semanal del grado de avance de las tareas respecto a la planificación para detectar retrasos tempranos.

- **Severidad (10):** La severidad del riesgo no cambia con la medida adoptada.
- **Ocurrencia (3):** Al detectarse un retraso en forma temprana se pueden aumentar los esfuerzos para cumplir con la planificación.

**Riesgo 2:**

**Plan de mitigación:** Se comprarán placas de desarrollo de los principales módulos.

- **Severidad (5):** Con las placas de desarrollo se pueden probar la mayoría de los requisitos.
- **Ocurrencia (5):** Al tener otra alternativa para probar el firmware la probabilidad de no poder probarlo baja.

**Riesgo 4:**

**Plan de mitigación:** Se planifican las tareas mas complejas y que mayor incertidumbre dan al comienzo del proyecto para tener margen de acción ante problemas para su ejecución.

- **Severidad (9):** La severidad del riesgo no cambia con la medida adoptada.
- **Ocurrencia (3):** Al detectarse un retraso en forma temprana se pueden aumentar los esfuerzos para cumplir con la planificación.

## 14. Gestión de la calidad

### 1. Requerimientos de interfaces externas

- 1.1. El software deberá comunicarse con el equipo de registro utilizando una interfaz BLE 5.0.

**Verificación:** Se verificarán las librerías y módulos de software utilizados.

**Validación:** Pruebas de sistema sobre el producto final.

- 1.2. El software deberá comunicarse con el equipo de registro utilizando una interfaz RS485.

**Verificación:** Se verificarán las librerías y módulos de software utilizados.

**Validación:** Pruebas de sistema sobre el producto final.

- 1.3. El software deberá indicar mediante el led1 que está transmitiendo las señales adquiridas de forma inalámbrica o cableada.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas. Inspección visual.

**Validación:** Pruebas de sistema sobre el producto final. Inspección visual.

- 1.4. El software deberá indicar mediante el led2 si está encendido o apagado. También deberá indicar con el mismo led2 si entra en el modo pairing BLE.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas. Inspección visual.

**Validación:** Pruebas de sistema sobre el producto final. Inspección visual.

- 1.5. El software deberá manejar una salida para sincronizar la adquisición con un estimulador externo (dispositivo externo como slave). La salida deberá poder configurarse entre normal bajo y normal alto. El pulso deberá poder configurarse entre 5 anchos de pulso diferentes (0,1 ms; 0,5 ms; 1 ms; 5 ms; 10 ms).

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Medición con osciloscopio sobre el producto final.

- 1.6. El software deberá manejar una entrada para sincronizar la adquisición con un estimulador externo (dispositivo externo como master). La detección deberá ser por flanco y se deberá poder configurar si el flanco es de subida o de bajada.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Se inyectará un pulso de sincronismo sobre el producto final.

- 1.7. El software deberá manejar el pulsador que servirá para encender el dispositivo y para realizar el pairing BLE.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Se probará el funcionamiento del pulsador.

- 1.8. El software deberá generar la señal de impedancia para los canales de potenciales evocados.

**Verificación:** Se medirá con osciloscopio la señal de impedancia.

**Validación:** Pruebas de sistema sobre el producto final. Se colocarán resistencias en los electrodos para medir la impedancia.



- 1.9. El software deberá medir el estado de las baterías con el ADC interno del MCU.

**Verificación:** Se conectará una fuente de tensión variable par verificar la medición de tensión de las baterías.

**Validación:** Pruebas de sistema sobre el producto final. Se leerá la indicación del software y se contrastará midiendo las baterías con un multímetro.

## 2. Requerimientos funcionales

- 2.1. El software deberá configurar la frecuencia de muestreo de la señal adquirida entre 7 diferentes valores ( el ADC de "EEG-FrontEnd\_v1.x.0.SchDoc" ofrece 65 Hz, 131 Hz, 262 Hz, 524 Hz, 1048 Hz, 2096 Hz y 4193 Hz).

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Se revisarán los archivos adquiridos por el producto final.

- 2.2. El software deberá poder adquirir de 1 a 8 canales simultáneos.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Inspección visual en el dispositivo de registro.

- 2.3. Mediante comandos recibidos por BLE 5.0 o RS485 el software deberá poder iniciar y parar la adquisición.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Inspección visual en el dispositivo de registro.

- 2.4. El software deberá configurar la ganancia de amplificación de cada canal entre 7 diferentes valores ( el ADC de "EEG-FrontEnd\_v1.x.0.SchDoc" ofrece 1, 2, 4, 6, 8, 12 y 24).

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final.

- 2.5. El software deberá configurar el ADC para usar una tipología de entrada diferencial o referencial.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.

**Validación:** Pruebas de sistema sobre el producto final. Se inyectará señal en las entradas según configuración seleccionada.

- 2.6. El software deberá medir la impedancia de los electrodos con una señal de medición de impedancia cuya frecuencia deberá ser de 7,8 Hz o 31,2 Hz.

**Verificación:** Se medirá con osciloscopio la señal de medición de impedancia.

**Validación:** Pruebas de sistema sobre el producto final.

- 2.7. El software deberá seleccionar a qué electrodos le inyecta la señal de medición de impedancia.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas. Se medirá con osciloscopio la señal de medición de impedancia.

**Validación:** Pruebas de sistema sobre el producto final. Inspección visual en el dispositivo de registro.

- 2.8. El software deberá seleccionar para cada electrodo si usa la señal de impedancia generada por el ADC o la generada por el MCU.

**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas. Se medirá con osciloscopio la señal de medición de impedancia.

**Validación:** Pruebas de sistema sobre el producto final. Se colocarán resistencias en los electrodos para medir la impedancia.

- 2.9. El software deberá enviar la siguiente información de autodiagnóstico: temperatura del ADC, valor de las tensiones del ADC y frecuencia del clock del ADC.  
**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.  
**Validación:** Pruebas de sistema sobre el producto final.
- 2.10. El software deberá prender y apagar el equipo con una pulsación corta del botón, pulsación menor a 1 segundo.  
**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.  
**Validación:** Pruebas de sistema sobre el producto final. Se probará el funcionamiento del botón.
- 2.11. El software deberá entrar en el modo de apareo de la comunicación BLE con una pulsación larga del botón, pulsación mayor a 4 segundos.  
**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.  
**Validación:** Pruebas de sistema sobre el producto final. Se probará el funcionamiento del botón.
- 2.12. Luego de 1 minuto de inactividad, el software deberá entrar a un modo de bajo consumo de energía. Para esto deberá apagar el ADC, el transceiver RS485 y la alimentación de todos los periféricos externos al MCU.  
**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.  
**Validación:** Pruebas de sistema sobre el producto final. Medición del consumo.
- 2.13. El software deberá manejar una comunicación RS485 hasta 3 MBd.  
**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.  
**Validación:** Pruebas de sistema sobre el producto final. Se probará la comunicación a 3MBd.
- 2.14. El software deberá poder recibir comandos mientras está transmitiendo las señales adquiridas.  
**Verificación:** Pruebas unitarias y de integración sobre las funciones asociadas.  
**Validación:** Pruebas de sistema sobre el producto final. Se enviarán comandos mientras se adquiere señal.

### 3. Requerimientos de documentación

- 3.1. La documentación debe cumplir los requisitos de la norma ISO 62304.  
**Verificación:** Por inspección, se completará un checklist.  
**Validación:** No esta dentro de los alcances del proyecto.
- 3.2. Toda la documentación del proyecto se almacenará bajo un sistema de control de versiones GIT.  
**Verificación:** Por inspección del repositorio.  
**Validación:** Por inspección del repositorio.
- 3.3. Toda la documentación del código se realizará utilizando la herramienta Doxygen.  
**Verificación:** Por inspección del código fuente.  
**Validación:** Por inspección de la página web generada.
- 3.4. Se deberá realizar un informe de avance del proyecto en el séptimo mes de trabajo  
**Verificación:** Por inspección del informe.  
**Validación:** Por inspección del informe.
- 3.5. Se deberá realizar la memoria técnica del trabajo final con la plantilla elaborada por la cátedra de gestión de proyecto.  
**Verificación:** Por inspección de la memoria.  
**Validación:** Por inspección de la memoria.

## 15. Procesos de cierre

Al finalizar el proyecto se realizarán las siguientes actividades:

1. Informe de evaluación final del proyecto, para lo cual se analizarán los siguientes aspectos:
  - Si se cumplió con la fecha de finalización del proyecto.
  - Si se tuvieron que utilizar más o menos recursos de los planificados para cumplir con el plan.
  - Si se pudo cumplir con todos los requisitos planificados.
  - Si se debieron tomar acciones no planificadas.
  - Si aparecieron riesgos no contemplados originalmente.
  - Si surgieron problemas y como se solucionaron.
  - Nivel de satisfacción del cliente.
2. Transferencia de los entregables al cliente.
  - Se coordinará una reunión con el cliente para tal fin.
3. Presentación pública en el marco de la CESE.
4. Acto de cierre.
  - Se realizará un acto de cierre con el objetivo de agradecer a todas las personas involucradas en el proyecto, miembros del jurado, docentes y autoridades de la CESE.

Todas las actividades del proceso de cierre estarán a cargo del responsable del proyecto.