



# Sistema IoT para el monitoreo de niveles de ruido en entornos cerrados o pequeñas áreas urbanas

Autor:

Ing. José Pedro Rivero Peña

Director:

Mgtr. Ing. Rodolfo Raúl Terán Aiza (UMSS)

*Esta planificación fue realizada en el curso de Gestión de proyectos entre el 04 de marzo de 2025 y el 22 de abril de 2025.*

## 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 . . . . .	7
5. Supuestos del proyecto. . . . .	8
6. Requerimientos . . . . .	8
7. Historias de usuarios ( <i>Product backlog</i> ). . . . .	10
Historias de Usuario. . . . .	10
8. Entregables principales del proyecto . . . . .	11
9. Desglose del trabajo en tareas . . . . .	12
Desglose del trabajo en tareas. . . . .	12
10. Diagrama de Activity On Node. . . . .	14
11. Diagrama de Gantt . . . . .	15
12. Presupuesto detallado del proyecto . . . . .	18
13. Gestión de riesgos . . . . .	18
14. Gestión de la calidad . . . . .	21
15. Procesos de cierre . . . . .	23

## Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	04 de marzo de 2025
1	Se completa hasta el punto 5 inclusive	20 de marzo de 2025
2	Se completa hasta el punto 9 inclusive	28 de marzo de 2025
3	Se completa hasta el punto 12 inclusive	06 de abril de 2025
4	Se completa el plan	11 de abril de 2025
5	Se corrigen observaciones	19 de abril de 2025

## Acta de constitución del proyecto

Buenos Aires, 04 de marzo de 2025

Por medio de la presente se acuerda con el Ing. José Pedro Rivero Peña que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará “Sistema IoT para el monitoreo de niveles de ruido en entornos cerrados o pequeñas áreas urbanas” y consistirá en la implementación de un prototipo de un sistema IoT para el monitoreo de niveles de ruido en entornos cerrados o pequeñas áreas urbanas. El trabajo tendrá un presupuesto preliminar estimado de 600 horas y un costo estimado de 240.000 ARS, con fecha de inicio el 04 de marzo de 2025 y fecha de presentación pública en octubre de 2025.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg  
Director posgrado FIUBA

Mgtr. Ing. Rodolfo Raúl Terán Aiza  
Director del Trabajo Final

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

El monitoreo de los niveles acústicos en entornos cerrados y pequeños espacios urbanos es un problema que no se puede dejar de lado en la sociedad actual. La exposición constante a altos niveles de ruido puede generar estrés, fatiga auditiva y afectar la concentración y productividad en oficinas, hospitales, aulas y parques. A pesar de la existencia de sistemas de medición de ruidos, estos suelen ser costosos, estacionarios o requieren intervención manual para la toma y análisis de datos. En este contexto, surge la necesidad de desarrollar soluciones tecnológicas accesibles, autónomas y escalables para la medición en tiempo real de los niveles de ruido ambiental.

Este proyecto propone el desarrollo de un sistema IoT capaz de monitorear el nivel de ruido en tiempo real, mediante sensores que recopilan la información y la transmiten a una plataforma en la nube. De este modo, los usuarios pueden acceder a los datos y gestionar alertas asociadas a parámetros definidos. A diferencia de soluciones tradicionales, esta propuesta se enfoca en la modularidad y accesibilidad, permitiendo su implementación en diversos entornos sin necesidad de grandes inversiones en infraestructura. La solución contempla el uso de sensores calibrados, microcontroladores con conectividad inalámbrica y un sistema de almacenamiento y visualización de datos basado en la nube, lo que facilita el acceso a la información desde cualquier dispositivo con conexión a internet. La propuesta se diferencia de las existentes al integrar alertas configurables, escalabilidad para expansión del sistema y la posibilidad de incorporar algoritmos de inteligencia artificial en futuras fases para la identificación de patrones de ruido y predicción de niveles sonoros.

El sistema propuesto consta de varios módulos funcionales. En primer lugar, los nodos sensores distribuidos miden el nivel de ruido en diferentes ubicaciones y procesan localmente los datos antes de enviarlos a través de una red de comunicación Wi-Fi en entornos con infraestructura disponible y LoRa en espacios abiertos sin acceso a redes tradicionales. Los datos recopilados son almacenados en la nube y pueden visualizarse en una interfaz web, donde los usuarios pueden consultar registros históricos, recibir alertas en tiempo real y tomar decisiones en función de la información proporcionada por el sistema. Además, la optimización energética es un aspecto clave del diseño, por lo que los dispositivos funcionarán con baterías de larga duración y, en casos específicos, podrán incorporar alimentación solar para garantizar una operación continua.

Uno de los principales desafíos del proyecto es garantizar la precisión y confiabilidad de las mediciones en distintos entornos, para lo que se ha considerado la incorporación de un segundo tipo de sensor para controlar factores ambientales como temperatura y humedad, que pueden influir en la propagación del sonido. Asimismo, se busca desarrollar un sistema que pueda expandirse fácilmente, permitiendo la incorporación de nuevos nodos de medición sin afectar el rendimiento general de la plataforma. La seguridad en la transmisión y almacenamiento de los datos también es un aspecto crítico, por lo que se emplearán protocolos de comunicación seguros como MQTT con cifrado.

Este emprendimiento personal busca proporcionar una herramienta útil para el monitoreo de ruido en espacios donde el control acústico es un factor clave para la calidad de vida. La facilidad de implementación y la flexibilidad del sistema permiten su aplicación en múltiples sectores, desde instituciones educativas y hospitales hasta gobiernos municipales interesados en gestionar la contaminación acústica en espacios públicos. Con una infraestructura abierta y escalable, este sistema puede evolucionar para integrar nuevas funcionalidades.

A continuación, se presenta el diagrama del sistema, en el que se ilustra la interacción entre sus diferentes componentes y el flujo de datos desde la captación hasta la visualización de los datos.

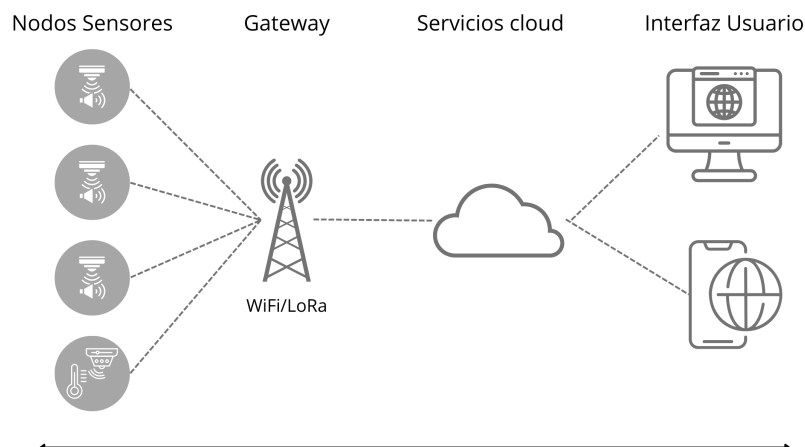


Figura 1. Diagrama del sistema.

## 2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Responsable	Ing. José Pedro Rivero Peña	FIUBA	Alumno
Orientador	Mgtr. Ing. Rodolfo Raúl Terán Aiza	UMSS	Director del Trabajo Final
Usuario final	Gobiernos, alcaldías, empresas	Pública y privada	-

- Orientador: el Mgtr. Rodolfo Raúl Terán Aiza cuenta con amplia experiencia en sistemas electromecánicos e instrumentación aplicada.

## 3. Propósito del proyecto

Desarrollar un sistema IoT eficiente, escalable y de bajo costo para la medición y monitoreo en tiempo real de los niveles de ruido en entornos cerrados y pequeñas áreas urbanas que permita recopilar, procesar y visualizar datos acústicos de manera accesible. Capaz de proporcionar a los usuarios información precisa sobre la contaminación sonora en distintos espacios.

#### 4. Alcance del proyecto

La propuesta está orientada a ofrecer un sistema modular, escalable y de bajo consumo energético, con una infraestructura adaptable a distintos escenarios de uso.

El proyecto incluye:

- **Diseño e implementación de un sistema IoT para monitoreo de ruido**, compuesto por nodos sensores, un módulo de comunicación y una plataforma en la nube.
  - Desarrollo de un prototipo funcional con tres nodos sensores distribuidos en un entorno de prueba.
  - Selección, calibración y configuración de sensores de ruido con un rango de medición entre 30 y 120 dB, que garantice una precisión aceptable para el análisis acústico.
  - Integración con microcontrolador esp32 que permita el procesamiento de datos en la transmisión de información.
  - Implementación de un módulo de conectividad inalámbrica utilizando tecnologías Wi-Fi para entornos cerrados y LoRa para áreas urbanas con baja cobertura de red.
- **Desarrollo de una plataforma en la nube para almacenamiento y visualización de datos.**
  - Configuración de una base de datos para el almacenamiento de registros históricos y análisis posterior.
  - Implementación de un *dashboard* web accesible para la consulta de datos en tiempo real, que permita la visualización de mediciones y tendencias históricas.
  - Incorporación de un sistema de alertas configurables para notificar a los usuarios cuando los niveles de ruido superen los umbrales predefinidos.
- **Optimización energética del sistema**, asegurando bajo consumo y autonomía en los nodos sensores.
  - Implementación de estrategias de bajo consumo en los microcontroladores, utilizando modos de suspensión y transmisión periódica.
  - Evaluación del uso de baterías recargables con posibilidad de integración de paneles solares en escenarios de monitoreo prolongado.

El presente proyecto no incluye:

- Reducción o mitigación del ruido ambiental, ya que su enfoque se limita a la medición y análisis de los niveles de ruido.
- Integración con modelos de inteligencia artificial o análisis predictivo en esta fase inicial, aunque el sistema está diseñado para futuras ampliaciones en este sentido.
- Desarrollo de una aplicación móvil nativa; el acceso a los datos y la visualización del sistema se realizarán únicamente a través de una interfaz web.
- Implementación en escenarios reales fuera del alcance de pruebas controladas, limitándose en esta etapa a un entorno de prueba definido para la validación del prototipo.

## 5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- No se considerarán regulaciones específicas sobre privacidad o normativas de manejo de datos, dado que el sistema solo recopilará información de niveles de ruido sin asociarlos a datos personales o identificables.
- Se asumirá que las condiciones de infraestructura en los entornos de prueba son representativas de espacios urbanos y cerrados típicos, sin interferencias extremas que afecten la captura y transmisión de datos.
- Se dispondrá de una red eléctrica estable en las áreas de prueba, que permita el funcionamiento de los sensores y la transmisión de datos sin interrupciones críticas.
- No se considerarán eventos extremos como fallos masivos de red, cortes prolongados de energía o interferencias radioeléctricas severas que puedan comprometer el funcionamiento del sistema.
- Se asumirá que las condiciones ambientales dentro de los entornos de prueba serán normales, sin presencia de factores climáticos extremos que puedan afectar la medición del sonido o el funcionamiento de los dispositivos.
- Se trabajará con un conjunto limitado de nodos sensores, sin escalabilidad masiva en esta fase del proyecto.
- No se contemplará la implementación de sistemas de redundancia avanzada para la transmisión y almacenamiento de datos, ya que el proyecto busca validar la funcionalidad básica del sistema en su fase inicial.

## 6. Requerimientos

A continuación, se enumeran los requerimientos del sistema propuesto.

### 1. Requerimientos funcionales (prioridad alta):

- 1.1. El sistema debe medir niveles de ruido ambiental en tiempo real con una frecuencia configurable (mínimo una medición cada 5 minutos).
- 1.2. Los sensores deben registrar valores en dB con un rango de 30–120 dB y precisión máxima de  $\pm 2$  dB.
- 1.3. El microcontrolador debe procesar localmente los datos y transmitirlos a la nube a través de Wi-Fi o LoRaWan, según el entorno.
- 1.4. El sistema debe generar una alerta automática cuando el nivel de ruido supere un umbral configurable.
- 1.5. El sistema debe permitir almacenar los datos de manera estructurada y persistente en una base de datos alojada en un servidor remoto.
- 1.6. El usuario debe poder acceder a los datos mediante un *dashboard* web desde cualquier navegador moderno.



- 1.7. El sistema debe estar probado en un entorno controlado como prueba de concepto, con documentación del despliegue y resultados.

## 2. Requerimientos de interfaz (prioridad alta):

- 2.1. El usuario debe poder acceder a un *dashboard* web desde navegadores modernos sin necesidad de instalar software adicional.
- 2.2. La interfaz debe mostrar los niveles de ruido actuales, valores históricos y tendencias de forma visual y comprensible (por ejemplo, gráficos de líneas, colores de alerta, etc.).
- 2.3. El usuario debe poder configurar el umbral de alertas desde la misma interfaz.

## 3. Requerimientos de hardware (prioridad alta):

- 3.1. Cada nodo debe operar al menos 5 días seguidos con batería sin recarga (con opción de carga solar).
- 3.2. Los sensores deben ser fácilmente reemplazables o reubicables.
- 3.3. El sistema debe funcionar en ambientes interiores y exteriores moderadamente protegidos.

## 4. Requerimientos de *backend* y almacenamiento (prioridad alta):

- 4.1. La base de datos utilizada deberá permitir el almacenamiento estructurado de registros históricos con acceso rápido (PostgreSQL o SQLite en entorno cloud).

## 5. Requerimientos de interoperabilidad (prioridad baja, opcional):

- 5.1. El sistema podrá exportar los datos en formato CSV o JSON para análisis externo (requerimiento opcional).

## 6. Requerimientos de seguridad (prioridad media):

- 6.1. El acceso al *dashboard* debe estar protegido mediante credenciales de usuario.
- 6.2. El *backend* debe implementar un sistema básico de autenticación, como JWT o similar, para proteger las rutas de acceso a datos.
- 6.3. Las comunicaciones entre el nodo y el servidor deben utilizar protocolos seguros, salvo en casos justificados por limitaciones técnicas.

## 7. Requerimientos normativos (prioridad alta):

- 7.1. El sistema debe contemplar los valores límite de exposición al ruido establecidos por la Ley N.º 1540 de la Ciudad Autónoma de Buenos Aires.
- 7.2. Las alertas deberán configurarse con base en dichos umbrales: 55 dB para interiores diurnos, 45 dB para nocturnos, 70 dB en espacios públicos al aire libre.

## 8. Requerimientos de documentación (prioridad alta):

- 8.1. Se debe desarrollar una memoria descriptiva del sistema, incluyendo el hardware, software y configuración utilizada.
- 8.2. Se debe incluir un manual de usuario y un manual de instalación y configuración.
- 8.3. La documentación debe contemplar información de APIs desarrolladas para la comunicación.

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

### Historias de Usuario

1. **Como administrador del sistema quiero visualizar en un panel web los niveles actuales de ruido en tiempo real para monitorear el estado acústico del entorno.**

*Story points:* 8 (complejidad: 3, dificultad: 2, incertidumbre: 2)

#### **Criterios de aceptación:**

- El *dashboard* debe mostrar los valores actuales de dB por nodo sensor.
- La información debe actualizarse automáticamente cada 5 minutos o menos.
- El diseño debe ser claro, responsivo y accesible desde cualquier navegador moderno.

2. **Como usuario del sistema quiero recibir una alerta cuando el nivel de ruido supere un umbral definido para poder tomar acciones correctivas.**

*Story points:* 8 (complejidad: 2, dificultad: 2, incertidumbre: 2)

#### **Criterios de aceptación:**

- El usuario podrá configurar el umbral desde el *dashboard*.
- El sistema debe generar la alerta a partir de la configuración de umbrales del usuario.
- La alerta se despliega en el aplicativo web y notifica al usuario.

3. **Como administrador quiero que los datos de ruido se almacenen automáticamente en una base de datos en la nube para garantizar persistencia y análisis posterior.**

*Story points:* 8 (complejidad: 3, dificultad: 3, incertidumbre: 2)

#### **Criterios de aceptación:**

- Cada medición debe almacenarse con marca de tiempo y origen del sensor.
- El sistema debe guardar al menos 7 días consecutivos de datos históricos.
- La base de datos debe ser accesible desde el *backend* de forma eficiente.

4. **Como administrador del sistema quiero poder exportar los datos en formato CSV o JSON para su uso en otros análisis.**

*Story points:* 5 (complejidad: 2, dificultad: 1, incertidumbre: 2)

#### **Criterios de aceptación:**

- El *dashboard* debe tener un botón de exportación por rango de fechas.
- El archivo generado debe contener datos válidos y bien estructurados.
- Debe ser posible descargarlo desde cualquier navegador.

5. **Como usuario quiero poder iniciar sesión con credenciales para acceder de forma segura al sistema.**

*Story points:* 8 (complejidad: 2, dificultad: 2, incertidumbre: 2)

#### **Criterios de aceptación:**

- Debe existir una pantalla de *login* con usuario y contraseña.

- Las credenciales deben verificarse en el *backend* con JWT u otra solución segura.
- No se debe acceder al *dashboard* sin estar autenticado.

**6. Como administrador quiero poder ver el estado de los nodos sensores para verificar que estén funcionando correctamente.**

*Story points:* 8 (complejidad: 3, dificultad: 2, incertidumbre: 2)

**Criterios de aceptación:**

- El sistema debe mostrar el estado de conexión de cada nodo (activo/inactivo).
- Cada nodo debe tener un identificador único.
- El *dashboard* debe mostrar su ubicación aproximada o nombre de ubicación.

**7. Como técnico de campo quiero desplegar un nodo sensor y que este se conecte automáticamente al sistema para facilitar su instalación.**

*Story points:* 8 (complejidad: 2, dificultad: 2, incertidumbre: 2)

**Criterios de aceptación:**

- El nodo debe autoconectarse a la red y al *backend* al encenderse.
- El nodo debe enviar una primera medición para confirmar su activación.
- El proceso no debe requerir configuración manual avanzada.

## 8. Entregables principales del proyecto

A continuación, se enumeran los entregables que serán producidos como resultado del desarrollo del proyecto.

- Prototipo físico funcional del sistema IoT con al menos tres nodos sensores de ruido operativos.
- Código fuente del *firmware* para los microcontroladores, con comentarios y estructura comprensible.
- *backend* del sistema implementado en entorno cloud con endpoints documentados.
- *Dashboard* web funcional para la visualización de datos, alertas e historial de mediciones.
- Base de datos implementada y operativa, con estructura simple y datos históricos registrados durante las pruebas.
- Diagrama de bloques del sistema.
- Esquema básico de conexiones electrónicas.
- Manual de instalación y uso
- Memoria del trabajo final.

## 9. Desglose del trabajo en tareas

### Desglose del trabajo en tareas

A continuación, se presenta el desglose del trabajo en tareas (WBS) del proyecto, con estimación de horas por grupo y su vinculación directa con las historias de usuario (HU) definidas.

1. Planificación y diseño del sistema (60 h)
  - 1.1. Análisis detallado de requerimientos y casos de uso (10 h)
  - 1.2. Diseño del sistema y definición de arquitectura general (16 h)
  - 1.3. Selección de sensores y componentes electrónicos (13 h)
  - 1.4. Diseño del diagrama de bloques del sistema (6 h)
  - 1.5. Definición de estructura de base de datos y *backend* (15 h)
2. Desarrollo del *firmware* para nodos sensores (120 h)
  - 2.1. Configuración e integración de sensores de ruido con microcontrolador (25 h)
  - 2.2. Programación del procesamiento de señales de medición en dB (30 h)
  - 2.3. Implementación de modos de bajo consumo *deep sleep* (15 h)
  - 2.4. Transmisión de datos vía Wi-Fi y LoRa (30 h)
  - 2.5. Calibración básica con sonómetro de referencia (8 h)
  - 2.6. Documentación técnica del *firmware* (12 h)
3. *backend* y almacenamiento de datos (110 h)
  - 3.1. Configuración de base de datos (PostgreSQL) (15 h)
  - 3.2. Implementación del *backend* en servidor cloud (30 h)
  - 3.3. Desarrollo de API para recepción y consulta de datos (25 h)
  - 3.4. Gestión de usuarios y autenticación (JWT) (15 h)
  - 3.5. Pruebas de integración y validación (10 h)
  - 3.6. Documentación técnica del *backend* (15 h)
4. Desarrollo del *dashboard* web (115 h)
  - 4.1. Maquetado del *dashboard* e interfaz responsiva (15 h)
  - 4.2. Visualización de niveles de ruido (15 h)
  - 4.3. Indicadores visuales (15 h)
  - 4.4. Configuración de umbrales de alerta por el usuario (25 h)
  - 4.5. Función de exportación de datos (CSV/JSON) (10 h)
  - 4.6. Implementación de login y control de acceso (20 h)
  - 4.7. Pruebas funcionales y correcciones (7 h)
  - 4.8. Documentación técnica del *frontend* (8 h)
5. Despliegue del prototipo y validación (65 h)
  - 5.1. Montaje y prueba de nodos sensores en entorno de prueba (15 h)

- 5.2. Monitoreo del sistema en operación (20 h)
- 5.3. Análisis de resultados y comportamiento (15 h)
- 5.4. Documentación de la prueba de concepto (15 h)
- 6. Documentación y entregables (150 h)
  - 6.1. Redacción de la memoria del proyecto (70 h)
  - 6.2. Elaboración del manual de usuario (20 h)
  - 6.3. Preparación de presentación final (20 h)
  - 6.4. Revisión de redacción, corrección y estilo (10 h)
  - 6.5. Diagramas (bloques, instalación, conexiones, Gantt) (20 h)
  - 6.6. Organización de anexos y referencias (10 h)

**Cantidad total de horas: 620 h**

## 10. Diagrama de Activity On Node

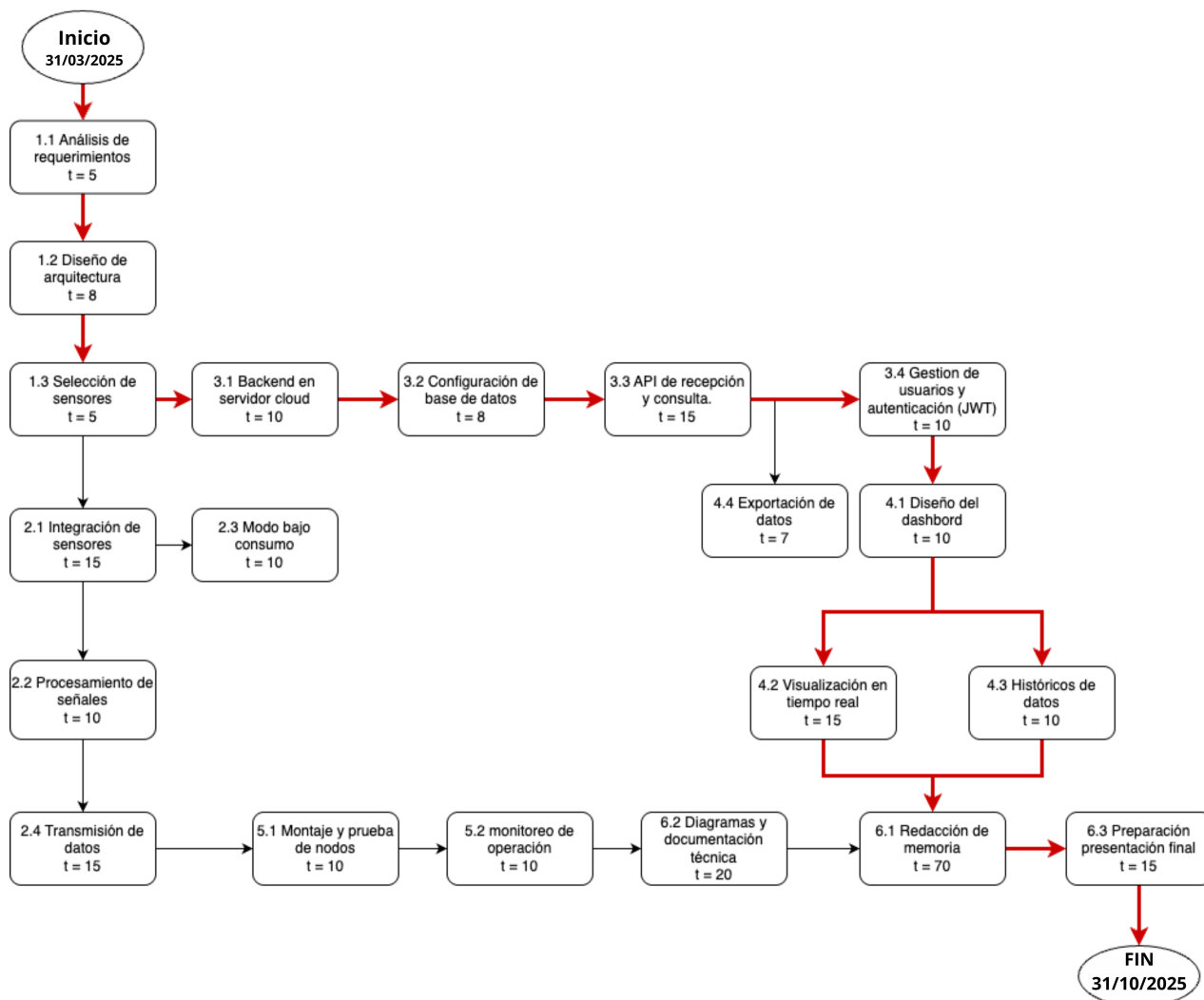


Figura 2. Diagrama de *Activity on Node*.

- Las líneas más gruesas de color rojo representan la ruta crítica del proyecto.
- Las duraciones de las tareas están expresadas en horas.

## 11. Diagrama de Gantt

En la figura 3 se muestra el diagrama de Gantt a nivel general del proyecto.

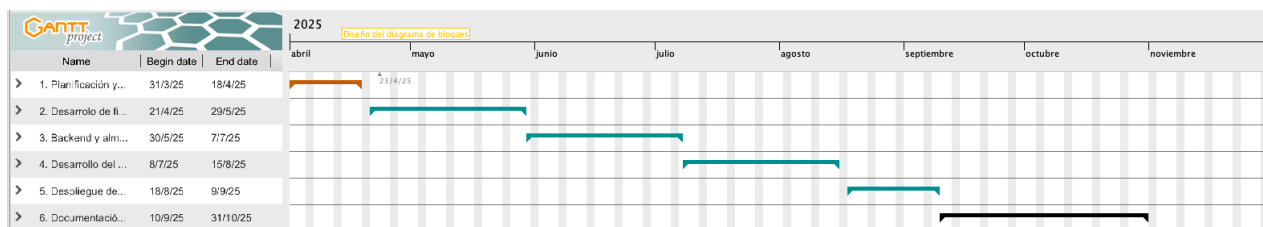


Figura 3. Diagrama de Gantt a nivel general del proyecto.

En la figura 4 se muestra el diagrama de Gantt para la planificación y diseño del sistema.

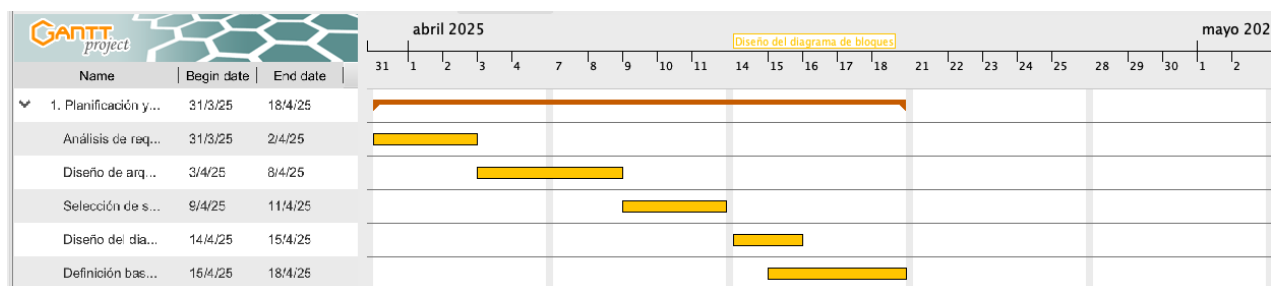


Figura 4. Diagrama de Gantt etapa 1.

En la figura 5 se muestra el diagrama de Gantt para el desarrollo del *firmware* para nodos sensores.

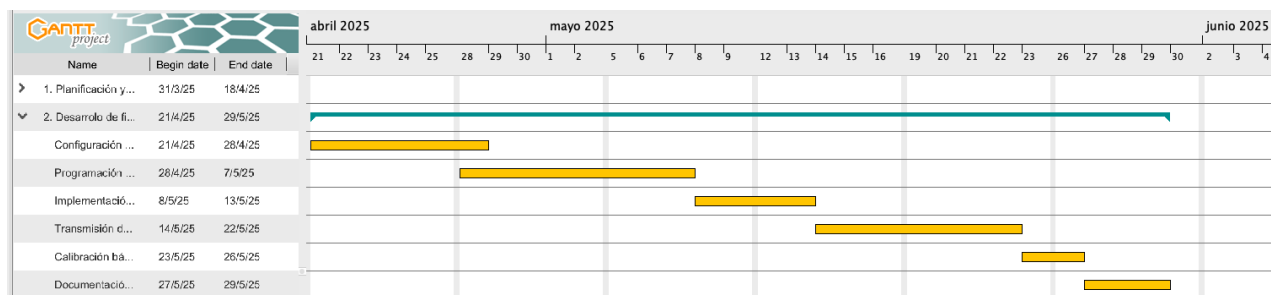


Figura 5. Diagrama de Gantt etapa 2.

En la figura 6 se muestra el diagrama de Gantt para el desarrollo de *backend* y almacenamiento de datos.

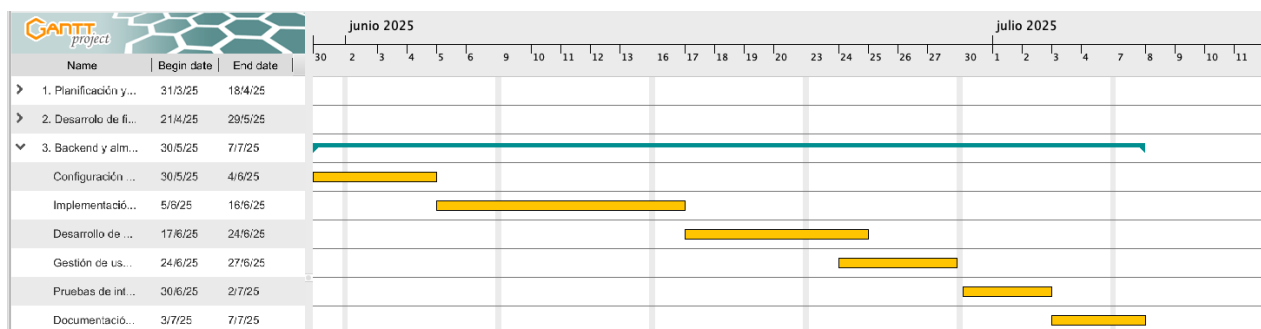


Figura 6. Diagrama de Gantt etapa 3.

En la figura 7 se muestra el diagrama de Gantt para el desarrollo del Desarrollo del *dashboard* web.

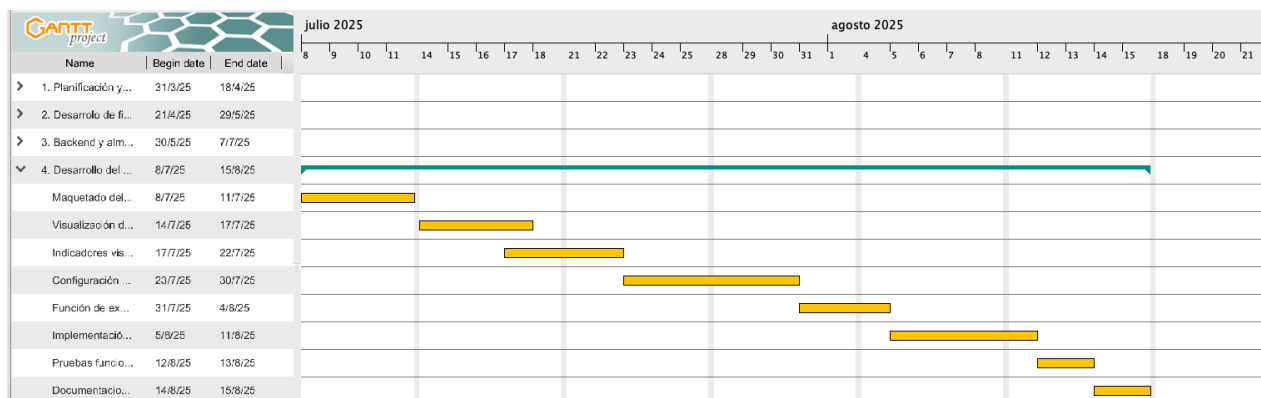


Figura 7. Diagrama de Gantt etapa 4.



En la figura 8 se muestra el diagrama de Gantt para el despliegue del prototipo y validación.

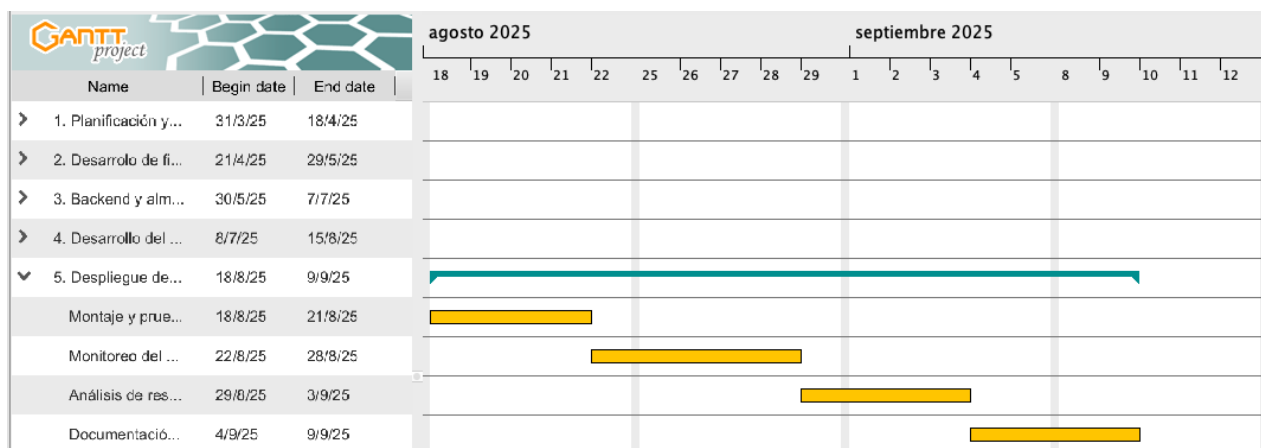


Figura 8. Diagrama de Gantt etapa 5.

En la figura 9 se muestra el diagrama de Gantt para el desarrollo de la documentación y entregables.

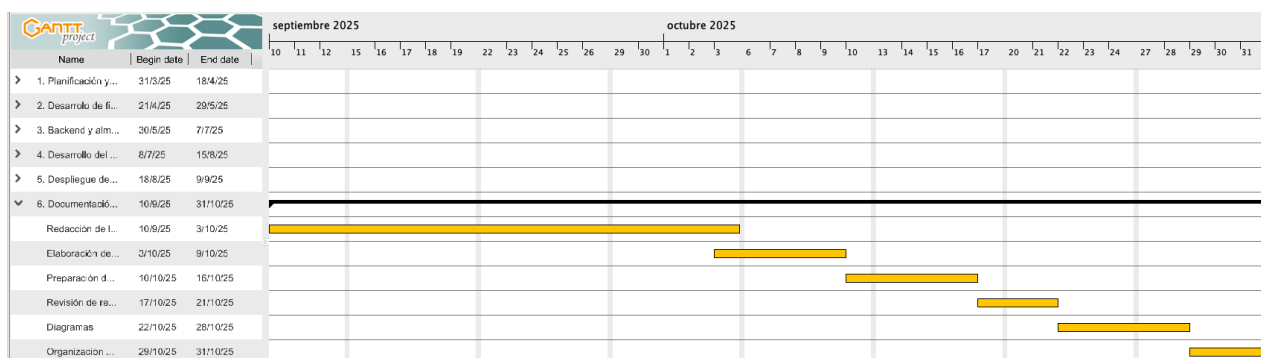


Figura 9. Diagrama de Gantt etapa 6.

## 12. Presupuesto detallado del proyecto

Los valores están expresados en pesos argentinos (ARS), cotización abril 2025.

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Sensores de ruido (micrófono electret + módulo amplificador)	3	ARS 8.000	ARS 24.000
ESP32 con LoRa	3	ARS 17.000	ARS 51.000
Caja protectora para cada nodo	3	ARS 10.000	ARS 30.000
Base de datos PostgreSQL (uso limitado - estimado equivalente)	1	ARS 2.000	ARS 2.000
Sensor de temperatura/humedad para compensación acústica (DHT22)	1	ARS 12.000	ARS 12.000
Gateway LoRa (alternativa: WisGate Edge Lite 2)	1	ARS 85.000	ARS 85.000
Componentes electrónicos varios (resistencias, jumpers, protoboards)	1	ARS 10.000	ARS 10.000
Horas de trabajo	620	ARS 10.000	ARS 6.200.000
SUBTOTAL			ARS 6.414.000
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Impresión de memoria y presentación	1	ARS 15.000	ARS 15.000
Transporte / imprevistos	1	ARS 4.000	ARS 4.000
30 % sobre los costos directos	1	ARS 1.924.200	ARS 1.924.200
SUBTOTAL			ARS 1.943.200
TOTAL			ARS 8.347.200

Tabla 1. Presupuesto del proyecto.

## 13. Gestión de riesgos

### a) Identificación de los riesgos y estimación de sus consecuencias:

A continuación, se identifican y analizan cinco riesgos relevantes que podrían afectar el desarrollo del proyecto. Para cada riesgo se estima su severidad (S) y su probabilidad de ocurrencia (O), en una escala del 1 al 10. Estos valores permiten priorizar los riesgos y tomar decisiones preventivas.

Riesgo 1: fallo en la conectividad de red (Wi-Fi o LoRa) durante la prueba del prototipo.

- Severidad (S): 8.  
Un fallo en la transmisión de datos impide la validación del sistema, lo que comprometería seriamente los resultados del proyecto.
- Ocurrencia (O): 5.  
Si bien se cuenta con infraestructura básica de red, existen riesgos asociados a interferencias, cobertura limitada o errores de configuración.

Riesgo 2: retraso en la entrega o disponibilidad de componentes electrónicos.

- Severidad (S): 6.  
El proyecto depende de componentes específicos (sensores, ESP32, *gateway*). La falta de alguno de ellos retrasaría el cronograma.
- Ocurrencia (O): 5.  
Los componentes seleccionados son comunes y disponibles en el mercado local, pero pueden existir demoras logísticas o rotura de stock.

Riesgo 3: fallos en la calibración de sensores de ruido.

- Severidad (S): 6.  
Una calibración incorrecta puede generar datos inexactos, afectando la utilidad del sistema.
- Ocurrencia (O): 6.  
La calibración requiere instrumentos de referencia (sonómetro) y condiciones controladas. No es trivial si hay ruido ambiental o interferencias.

Riesgo 4: problemas de compatibilidad o errores en la integración entre *firmware*, *backend* y *dashboard*.

- Severidad (S): 7.  
Una integración fallida impide la operación del sistema completo y su validación como solución IoT funcional.
- Ocurrencia (O): 5.  
Aunque se usarán tecnologías probadas y bien documentadas, la integración entre sistemas heterogéneos siempre presenta desafíos.

Riesgo 5: falta de tiempo para completar adecuadamente la documentación y preparación final.

- Severidad (S): 4.  
Un documento incompleto o una presentación pobre puede afectar la evaluación del trabajo, más allá del funcionamiento técnico.
- Ocurrencia (O): 7.  
Existe el riesgo de que se priorice la parte técnica, dejando la redacción para el final. Es un patrón común en proyectos similares.

## b) Tabla de gestión de riesgos:

Criterio adoptado:

Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores o iguales a 30.

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

Riesgo	S	O	RPN	S*	O*	RPN*
Fallo en la conectividad de red (Wi-Fi o LoRa).	8	5	40	6	3	18
Retraso en la entrega o disponibilidad de componentes electrónicos.	6	5	30	5	3	15
Fallos en la calibración de sensores de ruido.	6	6	36	5	3	15
Problemas de compatibilidad o errores en la integración entre <i>firmware</i> , backend y <i>dashboard</i> .	7	5	35	5	3	15
Falta de tiempo para completar adecuadamente la documentación y preparación final.	4	7	28	-	-	-

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

Se desarrollan a continuación los planes de mitigación correspondientes a los riesgos cuyo índice ( $S \times O$ ) iguala o supera el umbral establecido de 30.

Riesgo 1: fallo en la conectividad de red

Plan de mitigación: se definirá desde el inicio una doble vía de comunicación para los nodos (Wi-Fi y LoRa), con pruebas independientes de conectividad para ambos entornos. Se establecerán mecanismos de almacenamiento local de datos ante fallos temporales de red.

- Severidad ( $S^*$ ): 6.  
La severidad disminuye al incorporar redundancia de conectividad y mecanismos de respaldo que aseguran la recuperación de datos.
- Probabilidad de ocurrencia ( $O^*$ ): 3.  
La ocurrencia baja al realizar pruebas previas de cobertura y configurar adecuadamente las redes en el entorno de validación.

Riesgo 2: retraso en la entrega o disponibilidad de componentes

Plan de mitigación: se adelantará la compra de los componentes críticos al comienzo del proyecto. Se identificarán proveedores alternativos y se preverán equivalentes compatibles por si alguno no está disponible.

- Severidad ( $S^*$ ): 6.  
Aunque un retraso puede afectar la ejecución, contar con alternativas permite mantener la continuidad del desarrollo.
- Probabilidad de ocurrencia ( $O^*$ ): 2.  
Se reduce significativamente al anticipar las compras y contar con opciones de reemplazo técnico.

Riesgo 3: fallos en la calibración de sensores

Plan de mitigación: se utilizará un sonómetro clase 2 como referencia para la calibración de todos los sensores antes de su despliegue. Se documentará el procedimiento y se repetirá si hay cambios ambientales significativos.

- Severidad (S\*): 5.  
Con una calibración controlada, el impacto de una desviación se reduce a un margen aceptable.
- Probabilidad de ocurrencia (O\*): 3.  
La ocurrencia disminuye al aplicar un proceso sistemático y validado para todos los nodos.

#### Riesgo 4: errores en la integración de sistemas

Plan de mitigación: se establecerán pruebas de integración parciales en cada sprint. Se documentarán interfaces con claridad (API, formatos de datos) y se aplicarán herramientas de control de versiones y entornos virtuales.

- Severidad (S\*): 5.  
El riesgo de fallo completo se reduce, ya que los errores se detectan de forma temprana y localizada.
- Probabilidad de ocurrencia (O\*): 3.  
Se reduce mediante pruebas constantes y control estricto del ciclo de desarrollo.

## 14. Gestión de la calidad

Estas acciones aseguran que el sistema cumple con los criterios funcionales, técnicos y de uso esperados, considerando tanto el punto de vista técnico como la experiencia del usuario final.

- Req #1: El sistema debe medir niveles de ruido en tiempo real cada 5 minutos.
  - Verificación: se analizará el *firmware* del nodo sensor para confirmar el uso de temporizadores y ciclos de lectura periódicos cada 300 segundos.
  - Validación: se dejará el nodo funcionando durante una hora y se revisará en el *dashboard* que los datos se actualicen al menos 12 veces con intervalos regulares.
- Req #2: El sensor debe medir en un rango de 30 a 120 dB con una precisión de  $\pm 5$  dB.
  - Verificación: se revisará la hoja de datos del sensor y se realizará una calibración comparativa con un sonómetro.
  - Validación: se expondrá el nodo a diferentes fuentes de ruido y se validará que los valores mostrados sean coherentes con el entorno.
- Req #3: Los datos deben almacenarse en la base de datos en la nube.
  - Verificación: se inspeccionará el código del backend para confirmar que la inserción en base de datos se realiza correctamente vía API.
  - Validación: se generarán datos de prueba desde un nodo y se consultará manualmente la base para verificar que fueron registrados.
- Req #4: El sistema debe generar una alerta visual cuando se supere un umbral de ruido.
  - Verificación: se evaluará la función condicional de alerta en el código del *frontend* y se probará el diseño visual en ambiente de desarrollo.

- Validación: se simulará un valor superior al umbral y se pedirá al usuario que identifique si la alerta fue clara y notoria.
- Req #5: El usuario debe poder visualizar los datos desde un *dashboard* web.
  - Verificación: se verificará que el *frontend* se despliegue correctamente en un entorno local y se integre con el *backend*.
  - Validación: se pedirá a un usuario externo que acceda al sistema y confirme que puede consultar los datos sin asistencia técnica.
- Req #6: El acceso al sistema debe estar protegido mediante autenticación por credenciales.
  - Verificación: se revisará la implementación del sistema de autenticación (JWT) en el *backend* con verificación de tokens.
  - Validación: se realizará múltiples registros para ingresar con sus respectivas credenciales y se simularán accesos no autorizados.
- Req #7: El nodo debe operar al menos 5 días sin recarga.
  - Verificación: se calculará el consumo estimado en modo deep sleep y transmisión periódica con base en hojas de datos y pruebas reales.
  - Validación: se dejará un nodo funcionando con batería durante 5 días en modo real y se registrará su autonomía.
- Req #8: El sistema debe exportar los datos históricos en CSV.
  - Verificación: se probará la ruta de exportación y se verificará el contenido del archivo generado con datos de prueba.
  - Validación: se solicitará al cliente descargar un archivo y confirmar que contiene la información esperada y en formato comprensible.
- Req #9: El prototipo debe estar documentado con manual de uso e instalación.
  - Verificación: se revisará que el manual incluya diagramas, instrucciones de montaje y configuración paso a paso.
  - Validación: se pedirá a un tercero no involucrado que instale el nodo con base en el manual, a fin de comprobar si puede completar el proceso sin asistencia.
- Req #10: El *dashboard* debe mostrar correctamente la ubicación y estado de cada nodo.
  - Verificación: se comprobará que el *backend* identifique a cada nodo con un ID único y que esa información llegue al *frontend*.
  - Validación: se simularán desconexiones y se verificará que el usuario visualice correctamente el estado del nodo (activo/inactivo).

## 15. Procesos de cierre

A continuación, se describen las pautas previstas para garantizar una finalización ordenada y reflexiva del proyecto.

- **Análisis del cumplimiento del Plan de Proyecto:**
  - Responsable: Ing. José Pedro Rivero Peña
  - Procedimiento: se llevará a cabo una revisión detallada del Plan de Proyecto original, con el objetivo de:
    - Verificar si las tareas planificadas fueron ejecutadas en los tiempos previstos según el diagrama de Gantt.
    - Evaluar si los objetivos generales y específicos del proyecto se alcanzaron de forma completa y satisfactoria.
    - Confirmar si los requerimientos funcionales, técnicos y visuales definidos en la etapa inicial fueron cumplimentados y validados correctamente.
    - Comparar las historias de usuario desarrolladas con las planificadas en el backlog para medir el grado de cobertura funcional real.
- **Identificación de técnicas, problemas y soluciones:**
  - Responsable: Ing. José Pedro Rivero Peña
  - Procedimiento: Dado que se trata de un emprendimiento personal, se realizará un análisis retrospectivo documentado por el autor con el fin de:
    - Registrar los principales problemas que surgieron durante el desarrollo del proyecto, tanto técnicos como organizativos.
    - Detallar las soluciones implementadas y las decisiones de diseño que permitieron sortear los obstáculos.
    - Evaluar la efectividad de las técnicas y metodologías utilizadas, incluyendo selección de hardware, frameworks de desarrollo, flujo de trabajo y herramientas de integración.
    - Redactar un informe de lecciones aprendidas, destacando las buenas prácticas, las decisiones acertadas y las áreas en las que podría haberse optimizado el trabajo. Este documento servirá como base para futuros desarrollos o iteraciones del sistema.
- **Acto de cierre y agradecimientos:**
  - Responsable: Ing. José Pedro Rivero Peña
  - Procedimiento: Se organizará un acto de cierre institucional-académico conforme a lo previsto en el plan del trabajo final. Este proceso incluirá:
    - Defensa pública del proyecto: se realizará una presentación formal del sistema desarrollado, sus objetivos, resultados alcanzados, validaciones realizadas y conclusiones. Se prevé realizarla de forma virtual, facilitando la participación de los evaluadores y del público interesado.
    - Agradecimientos formales:
      - ◇ A las autoridades y docentes de la carrera de Especialización en Internet de las Cosas de la Universidad de Buenos Aires, por el acompañamiento académico y la formación recibida.

- ◇ Al director del trabajo final y quienes colaboraron de manera técnica o logística durante el desarrollo del sistema, con conocimientos específicos o asistencia en fases específicas del proyecto.
- ◇ A la institución universitaria que brindó el marco formativo y metodológico necesario para llevar adelante esta propuesta.
- ◇ Al tribunal evaluador, por su tiempo y el valor de su evaluación crítica y constructiva.
- Financiamiento: cualquier gasto asociado al cierre del proyecto (materiales de presentación, soporte digital, conectividad, impresión) será cubierto directamente por el autor.