



Sistema de monitoreo de calidad del aire

Autor:

Ing. Rodrigo Jurgen Pinedo Nava

Director:

Por definir (pertenencia)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 4 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	7
3. Propósito del proyecto	7
4. Alcance del proyecto	7
5. Supuestos del proyecto.	8
6. Requerimientos	8
7. Historias de usuarios (<i>Product backlog</i>).	10
8. Entregables principales del proyecto	13
9. Desglose del trabajo en tareas	13
10. Diagrama de Activity On Node.	15
11. Diagrama de Gantt	16
12. Presupuesto detallado del proyecto	17
13. Gestión de riesgos	17
14. Gestión de la calidad	20
15. Procesos de cierre	21

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	4 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 11 inclusive	4 de abril de 2025

Acta de constitución del proyecto

Buenos Aires, 4 de marzo de 2025

Por medio de la presente se acuerda con el Ing. Rodrigo Jurgen Pinedo Nava que su Trabajo Final de la Carrera de Especialización en Internet de las Cosas se titulará “Sistema de monitoreo de calidad del aire” y consistirá en la implementación de una red de sensores especializados para la medición de partículas en suspensión, dióxido de carbono, compuestos orgánicos volátiles, temperatura y humedad ambiental para la recopilación y transmisión de datos en tiempo real. El trabajo tendrá un presupuesto preliminar estimado de 600 horas y un costo estimado de \$ 500, con fecha de inicio el 4 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

Ing. Jose Mauricio Vargas Nuñez
Alicorp

Por definir
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

La contaminación del aire es un problema crítico que afecta tanto a entornos urbanos como industriales, con consecuencias directas sobre la salud pública y el medio ambiente. En Argentina, la situación es alarmante. Según la Organización Mundial de la Salud (OMS), el aire en el país tiene una media anual de $13 \mu\text{g}/\text{m}^3$ de partículas PM2.5, superando en un 30 % el nivel considerado seguro por la organización. En Buenos Aires, esta media anual asciende a $14 \mu\text{g}/\text{m}^3$, lo que implica un 40 % por encima del límite recomendado. Estas cifras se traducen en consecuencias graves, como la muerte anual de 85 niños por enfermedades vinculadas a la contaminación del aire en Argentina.

Este proyecto nace como un emprendimiento personal con el propósito de monitorear la calidad del aire en entornos industriales y urbanos y proporcionar información clave para la toma de decisiones. Actualmente, muchas ciudades y empresas carecen de sistemas eficientes y accesibles para medir en tiempo real la calidad del aire, lo que dificulta la prevención y el control de la contaminación. El objetivo es llenar ese vacío con una solución tecnológica asequible y escalable.

Se propone desarrollar un sistema de monitoreo basado en una red de sensores y la tecnología internet de las cosas (IoT). El sistema será capaz de detectar partículas en el aire tales como PM2.5, niveles de CO₂, compuestos químicos dañinos, temperatura y humedad. Los datos obtenidos serán enviados a una plataforma accesible desde un aplicativo web. Los dispositivos estarán conectados mediante LoRaWAN y WiFi/MQTT, almacenarán datos de manera eficiente y se presentarán en una plataforma intuitiva. Revisar la Figura 1 para comprender el diagrama de bloques del sistema de monitoreo.

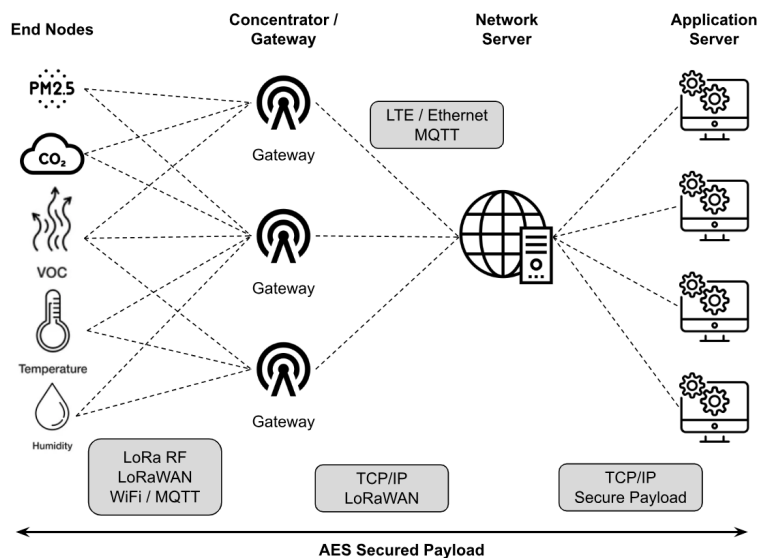


Figura 1. Diagrama en bloques del sistema.

Este proyecto está diseñado como una solución adaptable, principalmente para implementarse en empresas, con la capacidad de escalar a hogares y gobiernos. Todos los usuarios comparten una preocupación en común, la calidad del aire y su impacto en la salud. El sistema no solo permite el monitoreo en tiempo real, sino también envía alertas cuando los niveles de contaminación superan los límites recomendados. También ofrece herramientas para analizar datos históricos e identificar tendencias, que permitan tomar decisiones oportunas. Más que una simple innovación tecnológica, este proyecto representa una herramienta clave para mejorar la calidad de vida y promover un entorno más saludable y sostenible.

En el mercado actual existen soluciones para el monitoreo de la calidad del aire, sin embargo, muchas de ellas presentan limitaciones, como pueden ser:

- Costos elevados: en las etapas de implementación y mantenimiento, un sistema con características similares puede volverse solo accesible a instituciones con grandes presupuestos.
- Cobertura limitada: debido a que la mayoría de los sensores requieren cableado o dependencias de redes WiFi con alcance reducido.
- Falta de integración con plataformas accesibles: lo que dificulta el análisis y la interpretación de los datos por parte de usuarios sin conocimientos técnicos avanzados.

A diferencia de las soluciones mencionadas, este proyecto estará diseñado como una solución flexible y adaptable a empresas, hogares y gobiernos. Su propuesta de valor se basa en ofrecer un monitoreo ambiental accesible, para lograr entornos seguros y sostenibles, bajo los siguientes aspectos:

- Accesibilidad: una solución asequible en comparación con otros sistemas comerciales.
- Escalabilidad: implementación modular, adaptable a distintos entornos y necesidades.
- Interfaz intuitiva: plataforma accesible para cualquier usuario, sin necesidad de conocimientos técnicos avanzados.
- Conectividad eficiente: uso de tecnologías de comunicación de bajo consumo y gran alcance.
- Toma de decisiones informada: alertas y análisis de datos para implementar medidas de mitigación de contaminación.

El proyecto se encuentra en una etapa inicial de desarrollo. Para su primera versión se plantea una solución funcional, enfocada en validar su desempeño en entornos reales. En esta fase, el sistema ofrecerá dos modalidades de monitoreo que permitirán a los usuarios gestionar sus dispositivos de manera flexible:

- Monitoreo privado: cada usuario podrá registrar y gestionar sus propios dispositivos, accediendo a la información en tiempo real de los sensores vinculados.
- Monitoreo público: si el usuario así lo decide, podrá compartir los datos recopilados con la comunidad, permitiendo que la información esté disponible en una red abierta. Esto fomentará la creación de un ecosistema colaborativo.

En esta primera versión no se implementarán modelos de suscripción ni esquemas de pago, ya que el objetivo principal es desarrollar un prototipo funcional. Este proyecto permitirá evaluar la viabilidad técnica y el impacto del sistema en distintos escenarios de uso.

A futuro se integrará inteligencia artificial (IA) para análisis predictivo y se aumentarán funcionalidades para mejorar la toma de decisiones urbanas e industriales. El proyecto busca un crecimiento accesible y escalable, garantizando que cada usuario se beneficie de un monitoreo ambiental confiable desde su primera versión.

2. Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Ing. Jose Mauricio Vargas Nuñez	Alicorp	Supervisor de Almacén
Responsable	Ing. Rodrigo Jurgen Pinedo Nava	FIUBA	Alumno
Orientador	Por definir	pertenencia	Director del Trabajo Final
Usuario final	Industrias, hogares o gobiernos	Privada o pública	

- Cliente: el Ing. Jose Mauricio Vargas Nuñez es quien propuso los requerimientos del proyecto.
- Orientador: Por definires un profesional idóneo para la temática con especialidad en tecnologías IoT y LoRaWAN.

3. Propósito del proyecto

El propósito de este proyecto es fomentar entornos más saludables, seguros y sostenibles en zonas urbanas e industriales, a partir de una mejora en la calidad del aire. Esto se logrará mediante un sistema de monitoreo basado en IoT, que permita recopilar, analizar y visualizar datos ambientales en tiempo real. Con esta herramienta, se busca facilitar la identificación de fuentes de contaminación y alertar para que las partes interesadas tomen acciones correctivas oportunas. Además, se busca que la solución sea escalable, accesible y eficiente, lo que asegurará su adaptabilidad a distintos entornos y necesidades.

4. Alcance del proyecto

Este proyecto abarca el desarrollo e implementación de un sistema de monitoreo de calidad del aire basado en IoT. El sistema será capaz de medir, transmitir, almacenar y mostrar datos en tiempo real, proporcionando información clave para la toma de decisiones.

El proyecto incluye:

- Sensores para medir partículas en suspensión, dióxido de carbono, compuestos orgánicos volátiles, temperatura y humedad.
- Microcontrolares ESP32 con modulo LoRa y antena para una comunicación con el servidor mediante protocolos de LoRaWAN, WiFi, MQTT y TCP/IP.
- Servidor AWS IoT Core, usando *brocker* Mosquito y el diseño de la base de datos SQL/NoSQL con PostgreSQL.
- El diseño de una aplicación web para las visualizaciones del usuario, donde se verán los dispositivos conectados, mediciones, gráficas de históricos y alertas.

El presente proyecto no incluye:

- Desarrollo de modelos de suscripción o monetización, ya que en esta fase el enfoque es la validación del prototipo.
- Integración con inteligencia artificial o modelos predictivos avanzados.
- Implementación de una red de sensores a gran escala más allá del piloto inicial.
- Certificaciones oficiales de calidad del aire, ya que el sistema servirá como referencia complementaria a mediciones gubernamentales o institucionales.
- Localización geográfica que marque el estado de las zonas monitoreadas.

El alcance del proyecto está limitado a ser considerado un prototipo, enfocado en validar la viabilidad técnica y operativa. Las futuras versiones podrán incorporar mejoras basadas en los resultados de esta etapa.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- Disponibilidad financiera: dado que se trata de un emprendimiento personal, los costos del proyecto podrán ser cubiertos por el responsable del proyecto.
- Disponibilidad tecnológica: se contará con acceso a los sensores, microcontroladores y todos los componentes necesarios para la fabricación de los dispositivos IoT.
- Tiempo: el responsable cumplirá con la planificación propuesta, evitará retrasos y culminará el proyecto de manera satisfactoria.
- Infraestructura de comunicación: podrán realizarse pruebas en entornos controlados pertinentes para el correcto desarrollo del proyecto.
- Servidores cloud: se contará con espacios de prueba en AWS, los dominios y servicios para utilizar el *broker* Mosquito.

6. Requerimientos

1. Requerimientos funcionales:

- 1.1. El sistema permitirá la captura periódica de datos ambientales a través de sensores conectados a dispositivos IoT.
- 1.2. Los datos se transmitirán al servidor en tiempo real utilizando LoRaWAN o WiFi/MQTT según el entorno.
- 1.3. El sistema deberá almacenar los datos para su posterior análisis.
- 1.4. Se deberá ofrecer la opción de publicar los datos de manera privada o pública.
- 1.5. La interfaz será capaz de mostrar a los usuarios las mediciones de sus dispositivos y las zonas que tenga registrado un usuario.

- 1.6. Deberá generarse una alerta cuando los valores superen los umbrales definidos como peligrosos.
2. Requerimientos de infraestructura
 - 2.1. Se deberá contar con acceso a una red WiFi estable o cobertura LoRaWAN en las zonas de instalación de los dispositivos.
 - 2.2. El sistema requerirá un servidor local o en la nube con capacidad para recibir, procesar y almacenar datos provenientes de múltiples nodos.
 - 2.3. La base de datos utilizada deberá estar optimizada para el manejo de series temporales y para la gestión estructurada de usuarios y configuraciones.
 - 2.4. La alimentación de los dispositivos deberá realizarse mediante batería recargable o fuente USB, con posibilidad de alimentación solar para aplicaciones en exteriores.
3. Requerimientos de documentación
 - 3.1. Deberá documentarse el proceso de montaje del hardware, incluyendo la conexión de sensores al microcontrolador ESP32-S3 y la configuración de red.
 - 3.2. Se deberá incluir un manual de instalación y uso de la plataforma web.
 - 3.3. La documentación deberá contemplar las APIs utilizadas o desarrolladas para la comunicación entre dispositivos y servidor.
4. Requerimientos del entregable
 - 4.1. El entregable consistirá en un prototipo funcional que incluya al menos dos dispositivos que realicen las mediciones ambientales, un nodo IoT operativo, una plataforma web de visualización, un sistema de almacenamiento de datos y un módulo de alertas.
 - 4.2. El sistema deberá estar probado en un entorno controlado y documentado como prueba de concepto.
 - 4.3. Se desarrollará la memoria final del proyecto.
5. Requerimientos de la interfaz
 - 5.1. La plataforma deberá contar con una interfaz web responsiva para el monitoreo.
 - 5.2. La interfaz permitirá visualizar datos en tiempo real, acceder a históricos, gestionar dispositivos y configurar alertas.
 - 5.3. Los indicadores de calidad del aire deberán representarse mediante códigos de colores intuitivos y comprensibles.
 - 5.4. El sistema ofrecerá una interfaz de visualización con lecturas de datos ambientales de las zonas definidas como públicas. Se limitará la visualización al estado de las zonas sin tener acceso a los dispositivos vinculados y datos históricos.
6. Requerimientos funcionales del sistema para el rol “usuario”
 - 6.1. El usuario deberá poder registrarse y asociar dispositivos a su cuenta personal.
 - 6.2. El usuario gestionará sus dispositivos pudiendo agregar, eliminar, editar y asignar la ubicación en zonas definidas.
 - 6.3. Tendrá acceso a la visualización en tiempo real de los datos capturados por sus sensores.
 - 6.4. Podrá decidir si desea compartir sus datos de forma pública o mantenerlos en modo privado.

- 6.5. Recibirá alertas personalizadas cuando los valores medidos superen los umbrales definidos.
7. Requerimientos funcionales del sistema para el rol “administrador”
 - 7.1. El administrador deberá tener acceso a la gestión de usuarios, dispositivos y configuraciones generales del sistema.
 - 7.2. Tendrá visibilidad completa sobre las métricas generadas por todos los nodos activos.
 - 7.3. Deberá poder acceder a registros (*logs*) del sistema y supervisar el estado de funcionamiento de cada dispositivo.
8. Requerimientos funcionales del sistema para la vista pública
 - 8.1. La persona que quiera tener acceso a la vista pública deberá llenar un formulario donde se exprese su intención de acceder a esta información.
 - 8.2. La interfaz pública tendrá a disposición el estado en tiempo real de las mediciones ambientales de las zonas públicas.
9. Requerimientos de seguridad
 - 9.1. El sistema gestionará las credenciales para el ingreso de los usuarios.
 - 9.2. Se contará con métodos para recuperación de contraseñas y verificación de usuarios.
 - 9.3. La información asociada a los usuarios y a dispositivos configurados como privados deberá mantenerse protegida y no podrá hacerse pública sin consentimiento expreso.

7. Historias de usuarios (*Product backlog*)

Para estimar los story points de cada historia de usuario, se considerará la suma de tres factores: complejidad, dificultad e incertidumbre. Cada uno de estos factores será evaluado según una escala basada en la serie de Fibonacci, utilizando los siguientes valores de referencia:

- Muy bajo = 1
- Bajo = 2
- Medio = 3
- Alto = 5
- Muy alto = 8

Si el resultado de la suma es distinto a un número de la serie de Fibonacci, se le asignará el valor inmediatamente superior dentro de dicha secuencia. Esta metodología permitirá comparar de forma relativa el esfuerzo requerido entre las distintas historias, manteniendo una lógica coherente y adecuada al alcance del proyecto.

1. Como administrador del sistema, quiero ingresar a mi cuenta para gestionar la información de la cuenta, las zonas que deseo monitorear y los usuarios con el rol de supervisor.
Complejidad: 5 Dificultad: 5 Incertidumbre: 1 Suma: 11 → *Story Points*: 13
Criterios de aceptación

- El administrador con sus credenciales podrá tener acceso a su cuenta en la plataforma.
- El administrador podrá editar su información de la cuenta.
- El administrador puede gestionar las cuentas con el rol de supervisor.
- El administrador tendrá acceso a todas las funcionalidades del sistema.

2. Como usuario del sistema, quiero ingresar a mi cuenta que tiene rol de supervisor para gestionar mi información de la cuenta y monitorear las zonas asignadas.

Complejidad: 5 Dificultad: 5 Incertidumbre: 1 Suma: 11 → *Story Points*: 13

Criterios de aceptación:

- El usuario con sus credenciales podrá tener acceso a su cuenta en la plataforma.
- El usuario podrá editar su información de la cuenta.
- El usuario tendrá acceso a los tableros de monitoreo.

3. Como administrador del sistema, quiero registrar una zona de monitoreo en mi cuenta para poder acceder a sus datos ambientales.

Complejidad: 3 Dificultad: 2 Incertidumbre: 1 Suma: 6 → *Story Points*: 6

Criterios de aceptación:

- La zona quedará vinculada a la cuenta del usuario al ingresar su ID único.
- La zona aparecerá en el panel de la cuenta una vez registrado.
- El *backend* guarda la asociación en la base de datos y la valida en cada consulta.

4. Como usuario del sistema, quiero poder ver todas las zonas registradas para monitorear su estado general.

Complejidad: 3 Dificultad: 3 Incertidumbre: 3 Suma: 9 → *Story Points*: 13

Criterios de aceptación:

- Se listará el total de zonas registradas a monitorear.
- El estado de cada zona es visible mediante un ícono claro.
- El *backend* validará las zonas asignadas al usuario y en el *frontend* se enlistara.

5. Como administrador del sistema, quiero registrar un dispositivo IoT en mi cuenta para poder acceder a sus datos ambientales.

Complejidad: 5 Dificultad: 5 Incertidumbre: 3 Suma: 13 → *Story Points*: 13

Criterios de aceptación:

- El dispositivo quedará vinculado a la cuenta del usuario al ingresar su ID único.
- El dispositivo aparecerá en el panel de la cuenta una vez registrado.
- El *backend* guardará la asociación en la base de datos y la valida en cada consulta.

6. Como usuario del sistema, quiero poder ver todos los dispositivos registrados para monitorear su estado general.

Complejidad: 3 Dificultad: 3 Incertidumbre: 3 Suma: 9 → *Story Points*: 13

Criterios de aceptación:

- Se lista el total de dispositivos activos/inactivos.
- El estado de conexión de cada dispositivo es visible mediante un ícono claro.

- Los dispositivos se consultan mediante una API.

7. Como usuario, quiero visualizar los datos en tiempo real de mis sensores para conocer el estado del ambiente.

Complejidad: 5 Dificultad: 3 Incertidumbre: 3 Suma: 11 → *Story Points*: 13

Criterios de aceptación:

- Los datos se actualizan en intervalos definidos.
- Los valores se presentan en tarjetas con un diseño dashboard intuitivo.
- La plataforma consulta la API de datos en tiempo real.

8. Como usuario, quiero acceder al historial de datos recolectados para analizar la evolución de la calidad del aire.

Complejidad: 4 Dificultad: 3 Incertidumbre: 3 Suma: 10 → *Story Points*: 13

Criterios de aceptación:

- Se puede seleccionar un rango de fechas y consultar datos históricos.
- Los datos se grafican con líneas de tendencia y filtros por parámetro.
- La base de datos devuelve los datos en bloques optimizados para series temporales.

9. Como usuario, quiero configurar umbrales para cada sensor para recibir alertas ante niveles peligrosos.

Complejidad: 3 Dificultad: 2 Incertidumbre: 3 Suma: 8 → *Story Points*: 8

Criterios de aceptación:

- Se pueden definir umbrales distintos para cada parámetro.
- El usuario recibe un mensaje claro en la interfaz si se supera un umbral.
- Las reglas se almacenan por usuario y se validan en tiempo real.

10. Como usuario, quiero recibir una notificación en pantalla si se detecta contaminación peligrosa.

Complejidad: 2 Dificultad: 2 Incertidumbre: 1 Suma: 5 → *Story Points*: 5

Criterios de aceptación:

- El sistema envía automáticamente una notificación si se supera un umbral.
- El mensaje contiene el parámetro, el valor registrado y la hora.
- El envío se realiza mediante una alerta.

11. Como administrador, quiero poder elegir si los datos medidos con los dispositivos registrados son públicos o privados.

Complejidad: 2 Dificultad: 2 Incertidumbre: 1 Suma: 5 → *Story Points*: 5

Criterios de aceptación:

- El usuario puede alternar entre visibilidad pública y privada desde la configuración del dispositivo.
- Un ícono o leyenda indica claramente el estado actual.
- La API restringe el acceso a los datos privados a usuarios autenticados.

12. Como visitante del sistema, quiero ingresar a la plataforma utilizando una cuenta de invitado.

Complejidad: 3 Dificultad: 3 Incertidumbre: 3 Suma: 9 → *Story Points*: 13

Criterios de aceptación:

- El visitante podrá registrar sus datos requeridos para acceso al sistema.
- El visitante solo visualizará las zonas públicas.
- El *backend* restringirá el acceso a los datos privados.

13. Como visitante del sitio web, quiero poder visualizar los datos públicos de calidad del aire organizados por zonas para conocer el estado ambiental.

Complejidad: 5 Dificultad: 5 Incertidumbre: 3 Suma: 13 → *Story Points*: 13

Criterios de aceptación:

- El visitante tendrá acceso a las mediciones de los dispositivos públicos.
- El visitante llenara un formulario para para solicitar datos históricos.
- Los datos se actualizan con una frecuencia ya definida por defecto.

8. Entregables principales del proyecto

Los entregables del proyecto son:

- Aplicación web de visualización.
- Código fuente.
- Base de datos operativa.
- Manual de usuario.
- Manual de instalación.
- Esquemático de conexión de sensores.
- Diagrama de bloques del sistema.
- Memoria final del trabajo.

9. Desglose del trabajo en tareas

A continuación, se presenta el desglose de tareas estimadas para el desarrollo del proyecto:

1. Planificación (100 h)
 - 1.1. Estudiar sobre las mediciones para el monitoreo de la calidad del aire. (20 h)
 - 1.2. Investigar y seleccionar las tecnologías IoT a utilizarse en el proyecto. (20 h)
 - 1.3. Plantear el plan del proyecto. (60 h)

2. Ingreso y gestión de cuentas (70 h)
 - 2.1. Diseñar estructura de base de datos para usuarios y roles. (6 h)
 - 2.2. Desarrollar el *backend* de la API de *login* y autenticación. (8 h)
 - 2.3. Implementar la autenticación en el *frontend*. (8 h)
 - 2.4. Desarrollar formulario de inicio de sesión en *frontend*. (8 h)
 - 2.5. Diseñar vista de perfil de usuario con edición básica. (8 h)
 - 2.6. Desarrollar la lógica y datos del *backend* para gestionar supervisores. (8 h)
 - 2.7. Crear panel administrativo para gestionar supervisores. (8 h)
 - 2.8. Validar el proceso de ingreso y autenticación de usuarios al sistema. (8 h)
 - 2.9. Comprobar el control y la asignación de roles. (8 h)
3. Gestión de zonas (38 h)
 - 3.1. Diseñar modelo de datos para zonas en la base de datos. (6 h)
 - 3.2. Desarrollar API para CRUD de zonas. (10 h)
 - 3.3. Diseñar vista para registrar y listar zonas desde el *frontend*. (10 h)
 - 3.4. Validar la asignación de zonas a usuarios en *backend*. (6 h)
 - 3.5. Realizar pruebas de gestión de zonas. (6 h)
4. Gestión de dispositivos IoT (44 h)
 - 4.1. Diseñar esquema de relación entre dispositivos y zonas. (5 h)
 - 4.2. Crear formulario para registrar dispositivo desde el *frontend*. (7 h)
 - 4.3. Desarrollar API para vincular dispositivos al usuario/zona. (8 h)
 - 4.4. Mostrar listado de dispositivos vinculados y su estado. (8 h)
 - 4.5. Testear recepción de datos por MQTT desde dispositivo. (8 h)
 - 4.6. Realizar pruebas funcionales de gestión de dispositivos. (8 h)
5. Monitoreo de calidad del aire (50 h)
 - 5.1. Desarrollar lógica para almacenar datos sensados. (8 h)
 - 5.2. Crear API para consulta de datos en tiempo real e históricos. (8 h)
 - 5.3. Implementar tarjetas en *frontend* con datos de sensores. (10 h)
 - 5.4. Diseñar vista de gráficos históricos con filtros por fecha/zona. (12 h)
 - 5.5. Validar rendimiento y consumo de datos desde el *frontend*. (6 h)
 - 5.6. Realizar pruebas funcionales de las vistas de monitoreo. (6 h)
6. Alertas y configuración de umbrales (32 h)
 - 6.1. Agregar interfaz de configuración de umbrales por parámetro. (8 h)
 - 6.2. Programar *backend* para validación y activación de alertas. (8 h)
 - 6.3. Diseñar componente de notificación visual ante alertas. (8 h)
 - 6.4. Testear reglas con datos simulados en *backend*. (8 h)
7. Acceso y visibilidad de datos públicos (30 h)
 - 7.1. Diseñar API pública para consultar datos de zonas visibles. (6 h)

- 7.2. Crear vista pública con tarjetas de calidad del aire por zona. (10 h)
- 7.3. Implementar lógica para filtrar solo datos públicos. (6 h)
- 7.4. Agregar acceso como invitado y formulario básico. (4 h)
- 7.5. Validar acceso y permisos desde *frontend*. (4 h)
8. Despliegue y documentación técnica. (48 h)
 - 8.1. Configuración de la base de datos y definición de modelos. (8 h)
 - 8.2. *Deploy* de *backend* y *frontend* en plataforma gratuita. (8 h)
 - 8.3. Documentación técnica de la API. (8 h)
 - 8.4. Redacción del manual de usuario. (8 h)
 - 8.5. Manual de instalación del sistema. (8 h)
 - 8.6. Informe de pruebas del prototipo en entorno real. (8 h)
9. Documentación y presentación final. (195 h)
 - 9.1. Elaborar el informe de avance. (60 h)
 - 9.2. Desarrollar la memoria final con anexos y correcciones. (85 h)
 - 9.3. Armar la presentación final. (30 h)
 - 9.4. Ensayar y preparar la defensa. (20 h)

Cantidad total de horas: 607.

10. Diagrama de Activity On Node

En la figura 2 se visualiza el diagrama *Activity On Node* del proyecto:

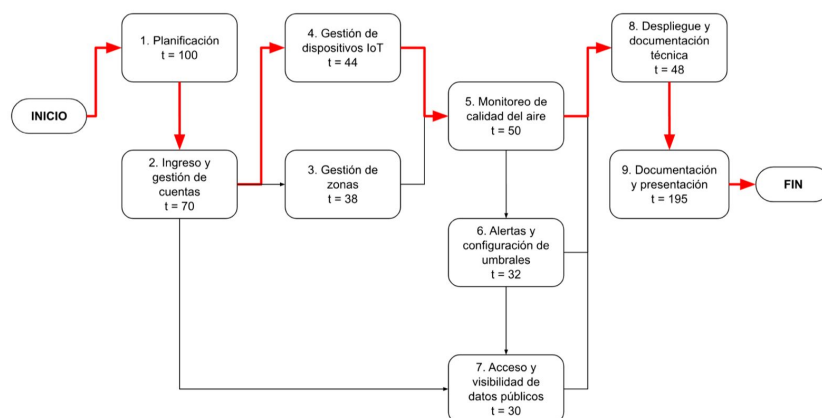


Figura 2. Diagrama de *Activity on Node*.

- Las tareas están agrupadas en etapas según el desglose del trabajo en tareas.
- Las duraciones t están expresadas en horas.
- Las flechas rojas indican caminos críticos parciales entre tareas dependientes.

11. Diagrama de Gantt

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

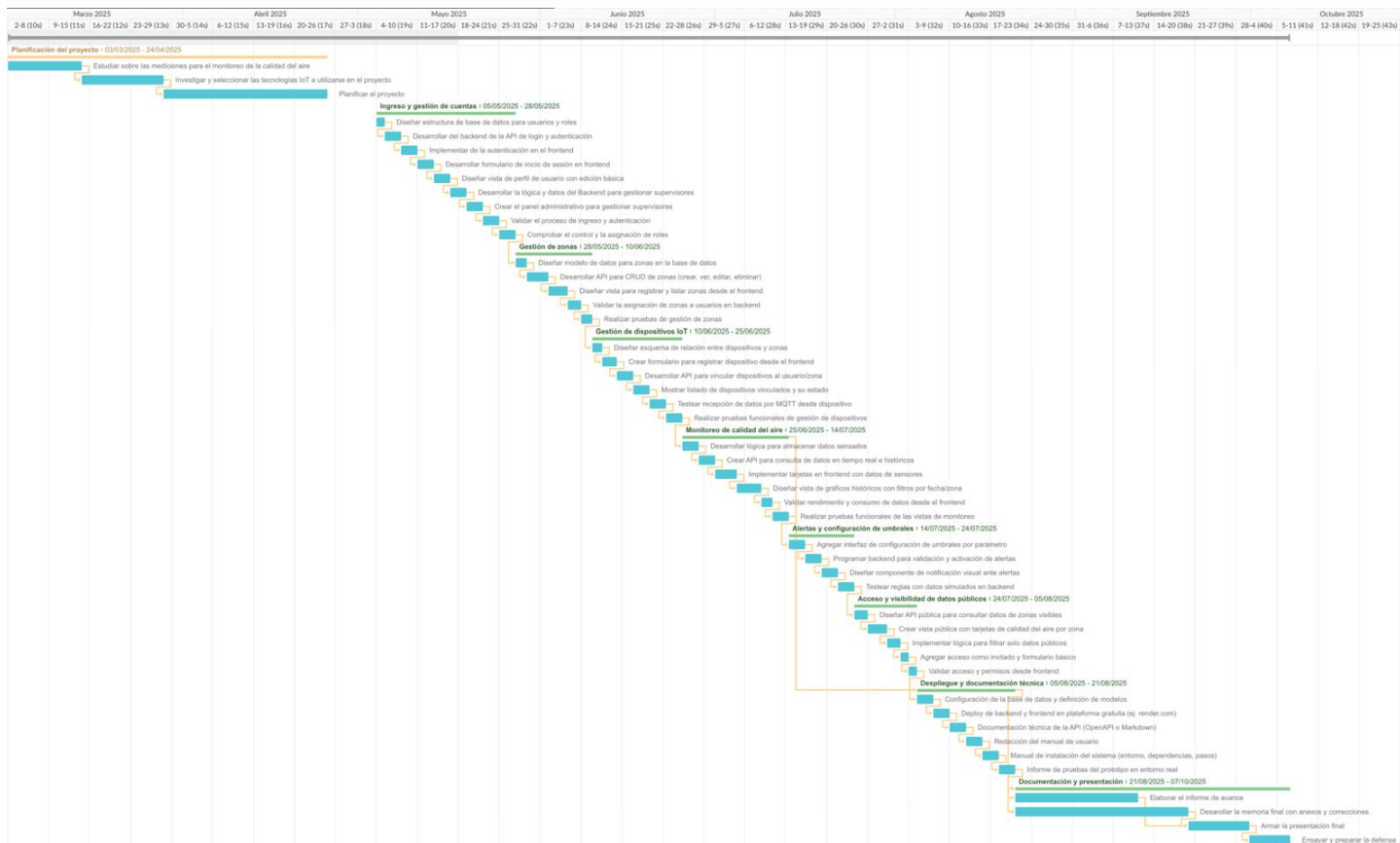


Figura 3. Diagrama de Gantt del proyecto.

12. Presupuesto detallado del proyecto

Los valores de los costos están expresados en dólares estadounidenses (USD). Al momento de elaborar el presente documento, 4 de abril del 2025, su tasa de conversión a pesos argentinos es de 1319/1 USD (referencia dólar MEP).

En la tabla 1, se visualiza un cuadro con el presupuesto del proyecto. Todas las unidades se encuentran expresadas en USD.

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario (USD)	Valor total (USD)
Sensor de polvo PM2.5 - PMS7003	3	\$27,00	\$81,00
Sensor de gas SCD41	3	\$31,00	\$93,00
Sensor CCS811 CO2	3	\$12,88	\$38,64
WisGate Edge Lite 2 RAK7268V2/RAK7268CV2	1	\$139,00	\$139,00
Mini UPS con salida 12VDC	1	\$19,00	\$19,00
ESP32 LoRa V3 + batería 1100 mAh + accesorios	4	\$31,50	\$126,00
Horas de trabajo	610	\$10,00	\$6.100,00
SUBTOTAL			\$6.596,64
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario (USD)	Valor total (USD)
30 % sobre costos directos	1	\$1.978,99	\$1.978,99
SUBTOTAL			\$1.978,99
TOTAL			\$8.575,63

Tabla 1. Presupuesto detallado del proyecto.

13. Gestión de riesgos

a) Identificación de los riesgos y estimación de consecuencias

Riesgo 1: fallo en la comunicación entre los dispositivos IoT y el servidor

- Severidad (S): 8
Una interrupción en la comunicación impide la recolección de datos en tiempo real, afectando la utilidad del sistema para el monitoreo ambiental.
- Ocurrencia (O): 6
Existe una probabilidad media-alta de ocurrencia debido a interferencias o problemas de cobertura en redes WiFi o LoRaWAN durante el prototipo.

Riesgo 2: sobrecarga de recursos en los dispositivos IoT

- Severidad (S): 8.
Si un nodo falla de forma inesperada, se pierde la continuidad del monitoreo, y puede interrumpirse la cadena de datos.

- Ourrencia (O): 6.

El ESP32, aunque potente, tiene limitaciones en RAM y CPU. Cargar múltiples sensores, comunicaciones paralelas y lógica de validación sin optimización puede llevar al cuelgue o reinicio del dispositivo.

Riesgo 3: problemas de compatibilidad entre sensores y microcontrolador

- Severidad (S): 6.

Podría afectar la calidad de los datos o requerir modificaciones de hardware/software no planificadas.

- Ourrencia (O): 5.

Se eligieron sensores compatibles, pero pueden presentarse inconsistencias en las librerías o el consumo de energía.

Riesgo 4: saturación o malformación de datos sensados

- Severidad (S): 7.

La mala interpretación de datos puede activar alertas falsas o provocar decisiones erróneas.

- Ourrencia (O): 6.

Es un riesgo común en sensores ambientales económicos o mal configurados.

Riesgo 5: fallo de los servicios de despliegue gratuito

- Severidad (S): 7.

Una interrupción del entorno de ejecución *cloud* o local durante una instancia crítica como pruebas, comprometería la validación del sistema en tiempo real.

- Ourrencia (O): 5.

Si bien los entornos pueden ser robustos, los errores de configuración, las restricciones presupuestarias en servicios *cloud* gratuitos, o incluso errores en el runtime pueden generar indisponibilidad momentánea.

b) Tabla de gestión de riesgos

Riesgo	S	O	RPN	S*	O*	RPN*
Fallo en comunicación IoT-servidor	8	6	48	6	4	24
Sobrecarga de recursos en los dispositivos IoT	8	6	48	6	4	24
Compatibilidad entre sensores y microcontrolador	6	5	30	5	4	20
Saturación o malformación de datos sensados	7	6	42	5	3	15
Fallo del servicio Render.com	7	5	35	5	3	15

Tabla 2. Evaluación de riesgos antes y después de la mitigación.

Criterio adoptado: Se tomarán medidas de mitigación para todos los riesgos con RPN inicial mayor a 25.

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

c) Plan de mitigación

Riesgo 1: fallo en la comunicación IoT-servidor

Plan de mitigación: incluir mecanismos de reintento automático, logs de errores y pruebas en distintos entornos para validar la conectividad. Asegurar compatibilidad entre MQTT y LoRaWAN mediante brokers intermedios.

- Severidad (S*): 6.
Se redujo al contar con validaciones de conectividad y almacenamiento temporal local.
- Ocurrencia (O*): 4.
Disminuye al hacer pruebas exhaustivas y usar hardware probado.

Riesgo 2: sobrecarga de recursos en los dispositivos IoT

Plan de mitigación: planificar tareas pequeñas, usar recursos de aprendizaje continuo y apoyo con herramientas de depuración. Reutilizar plantillas y librerías probadas.

- Severidad (S*): 6.
Con mecanismos de autorrecuperación, el impacto se reduce.
- Ocurrencia (O*): 4.
La probabilidad disminuye al optimizar el uso de memoria y mantener lógica desacoplada.

Riesgo 3: compatibilidad sensores - microcontrolador

Plan de mitigación: testeo unitario anticipado y validación con ejemplos oficiales de MicroPython y C/C++. Uso de fuentes de alimentación externa para sensores que lo requieran.

- Severidad (S*): 5.
El riesgo técnico se reduce con pruebas en laboratorio.
- Ocurrencia (O*): 4.
Menor al contar con documentación adecuada y soporte de comunidad.

Riesgo 4: saturación o malformación de datos sensados

Plan de mitigación: aplicar filtros de validación de rango en el *firmware* (por ejemplo, descartar valores extremos o “nan”), así como lógica de suavizado o *debounce* de señales antes de que lleguen al *backend*.

- Severidad (S*): 5.
Con validación previa se evita que el dato incorrecto afecte la plataforma.
- Ocurrencia (O*): 3.
La probabilidad se reduce si se validan lecturas en *firmware* + *backend*.

Riesgo 5: Fallo del servicio

Plan de mitigación: contar con una versión Dockerizada funcional del sistema que pueda ejecutarse localmente sin conexión a internet. Además, mantener backups de configuración en distintos proveedores y realizar ensayos de contingencia antes de la presentación.

- Severidad (S^*): 5.
El impacto se reduce al poder replicar la arquitectura localmente sin depender del servicio externo.
- Ocurrencia (O^*): 3.
La probabilidad disminuye al tener redundancia y pruebas previas del entorno final.

14. Gestión de la calidad

A continuación, se detallan los requerimientos considerados críticos o de alto valor para el cumplimiento del objetivo del sistema, junto con sus acciones de verificación y validación.

- Req # 1: el sistema deberá capturar datos ambientales a través de sensores conectados a dispositivos IoT.
 - Verificación: se probará cada sensor en condiciones controladas, comparando los valores con instrumentos de referencia o condiciones conocidas. Se revisará el *firmware* del ESP32 para verificar que realiza lecturas correctas.
 - Validación: se observará el comportamiento del sistema desde la interfaz web, verificando que los datos se actualizan en condiciones cambiantes (por ejemplo, al acercar una fuente de calor o CO₂).
- Req # 2: el sistema deberá enviar los datos al servidor en tiempo real utilizando WiFi o LoRa/MQTT
 - Verificación: se analizarán los paquetes enviados con herramientas como MQTT.fx o Mosquitto CLI, asegurando su frecuencia y estructura.
 - Validación: se revisará en el panel de usuario que los datos se actualicen en tiempo real tras un evento medible.
- Req # 3: El sistema deberá almacenar los datos para su posterior análisis.
 - Verificación: se inspeccionará directamente la base de datos para confirmar la correcta inserción de datos con marcas de tiempo y relaciones válidas.
 - Validación: se utilizará la interfaz para seleccionar un rango de fechas y verificar que se presentan correctamente los datos históricos.
- Req # 4: Se deberá ofrecer la opción de publicar los datos como privados o públicos.
 - Verificación: se revisará el *backend* para asegurar que se respeta el parámetro de visibilidad, y que la lógica de autorización limita el acceso a los datos según el estado de cada sensor o zona.
 - Validación: se probará la vista de invitado y se comprobará que no accede a datos privados.

- Req # 5: El sistema deberá generar alertas si se superan umbrales definidos como peligrosos.
 - Verificación: se forzarán lecturas fuera de rango para comprobar que la lógica de alertas se activa correctamente en el *backend*.
 - Validación: se observará en la interfaz que se muestra una alerta clara y comprensible, con el parámetro afectado y el valor medido.
- Req # 6: La plataforma deberá contar con una interfaz web responsiva.
 - Verificación: se realizarán pruebas en múltiples resoluciones usando herramientas de desarrollo y emuladores, verificando adaptabilidad de cada componente.
 - Validación: se accederá a la plataforma desde distintos dispositivos y se confirmará con el usuario que la experiencia de uso es adecuada.
- Req # 7: Los indicadores de calidad del aire deberán representarse mediante códigos de colores.
 - Verificación: se revisará la lógica del frontend para confirmar que los colores asignados se corresponden con los rangos establecidos en normativa.
 - Validación: el usuario verá un panel que indique visualmente, por color, el estado ambiental de cada zona.
- Req # 8: El usuario deberá poder registrar, editar y eliminar zonas de monitoreo.
 - Verificación: se probarán los *endpoints* de la API para operaciones CRUD en zonas y su impacto en la base de datos.
 - Validación: se confirmará, desde la interfaz, que se puede realizar la gestión de zonas sin errores, y que los cambios se reflejan inmediatamente.
- Req # 9: El sistema deberá ejecutarse en un entorno Dockerizado portátil.
 - Verificación: se construirá y ejecutará una imagen Docker del sistema, verificando su correcto funcionamiento desde CLI y logs del contenedor.
 - Validación: el contenedor se compartirá con un tercero para validar que el entorno puede reproducirse sin errores adicionales.
- Req # 10: El visitante podrá visualizar datos públicos sin necesidad de registrarse.
 - Verificación: se analizará el *backend* y las consultas para comprobar que no se requiere token y que sólo se retornan datos públicos.
 - Validación: como invitado, se accederá a la vista pública y se confirmará que los datos se muestran correctamente y están limitados a la visibilidad asignada.

15. Procesos de cierre

Una vez concluido el proyecto, se establecerán las siguientes pautas de trabajo que permitirán evaluar su ejecución, registrar aprendizajes y agradecer la colaboración de los distintos interesados:

- Análisis del cumplimiento del Plan de Proyecto original:

- A cargo del Ing. Rodrigo Jurgen Pinedo Nava, Ing. Jose Mauricio Vargas Nuñez y Por definir se revisará si las actividades definidas se completaron en los tiempos y alcances planificados. Se utilizará como referencia el diagrama de Gantt y los registros de avances documentados durante el desarrollo.
 - Se elaborará una síntesis comparativa entre lo planificado y lo realmente ejecutado, con observaciones sobre desviaciones y justificaciones correspondientes.
 - Este análisis será incluido en el informe final y presentado durante la reunión de cierre.
 -
- Identificación de técnicas útiles, problemas y soluciones aplicadas:
El Ing. Rodrigo Jurgen Pinedo Nava también será responsable de redactar un informe reflexivo donde se detallen:
- Las herramientas, tecnologías y metodologías que resultaron más eficaces para el desarrollo (por ejemplo, uso de Docker, modularización del backend, visualización en tiempo real).
 - Aquellas que no aportaron valor o que presentaron dificultades significativas.
 - Los principales problemas encontrados (como fallos en sensores o restricciones de conectividad) y las soluciones implementadas.

Esta documentación quedará como anexo del proyecto y será publicada junto al repositorio para referencia futura.

- Agradecimiento a interesados y colaboradores:
- El Ing. Rodrigo Jurgen Pinedo Nava se encargará de organizar el acto simbólico de agradecimiento a quienes participaron directa o indirectamente del proyecto, especialmente al director del trabajo final, al docente responsable, y a quienes brindaron apoyo técnico o institucional.
 - Esta actividad se realizará al finalizar la defensa del proyecto, incluyendo un breve mensaje público y mención especial en la memoria. No se contemplan gastos financieros significativos, por lo que no se requiere asignación presupuestaria específica.