

UNIVERSIDAD DE TARAPACÁ



FACULTAD DE INGENIERÍA

Departamento de Ingeniería en Computación e Informática



Plan de proyecto Detección de gases nocivos en plantas de Ariztia

Alumno(os):
Kamir Alfaro
Edson Galdames
Lukas Torres

Asignatura: Proyecto II

Profesor: Diego Aracena

ARICA, 28 Octubre 2025

Proyecto II

Formato para Plan de Proyecto

1.-Historial de Cambios

Fecha	Ver sió n	Descripción	Autor(es)
20/10 /2025	1.0	Versión preliminar del formato	Kamir Alfaro. Edson Galdames, Lukas Torres
27/10 /2025	1.1	Revisión y modificación del plan	Kamir Alfaro. Edson Galdames, Lukas Torres
24/11 /2025	1.2	Actualización con Etapa II del proyecto	Kamir Alfaro. Edson Galdames, Lukas Torres

Índice

1.-Historial de Cambios	2
2.- Panorama General	4
3.1.-Organización del proyecto	6
4.-Planificación de los procesos de gestión	7
4.1.-Planificación inicial del proyecto:	7
4.2.-Lista de Actividades:	9
Planificación del Proyecto	9
Ejecución del Proyecto	9
Cierre del Proyecto	10
4.3.-Planificación de la gestión de riesgos:	13
Avances del Proyecto	15
Evidencia Fotográfica	16
Plan de Integración	19
Planificación de los Procesos Técnicos	19
Planificación de los Procesos de Soporte	19
Especificaciones de Requerimientos	20
Requerimientos Funcionales	20
Requerimientos no Funcionales	20
Criterios de Aceptación	21
Diagrama de Clases	22
El diagrama de clases presentado modela la estructura lógica del sistema de monitoreo de gases implementado mediante sensores conectados a un ADC, con visualización a través de una interfaz gráfica. Cada clase representa un componente clave del funcionamiento del sistema y define sus atributos y métodos principales.	23
Clase ADC	23
Clase Alerta	23
Clase InterfazGUI	24
Diagrama de Secuencia	24
Secuencia general del proceso	25
Descripción de la Arquitectura del Sistema	27
1. Capa de Sensores (Modelo Físico–Electrónico)	27
2. Capa de Procesamiento (Modelo de Control)	27
3. Capa de Visualización (Modelo de Interfaz)	27
Conclusión Etapa II	29
Referencias	29

2.- Panorama General

Propósito:

El proyecto permitirá diseñar y construir un sistema automatizado para la detección de gases nocivos en los criaderos de pollos de Ariztía, con el fin de garantizar un ambiente seguro tanto para los trabajadores como para los animales, contribuyendo a la sostenibilidad y eficiencia operativa de la planta.

Alcance:

El sistema contará con sensores especializados (como los MQ-135 y MQ-137) para medir en tiempo real la concentración de gases como amoniaco (NH_3), metano (CH_4) y sulfuro de hidrógeno (H_2S).

Además, incluirá módulos de monitoreo y alerta (visual y/o sonora) que se activarán al superar los límites seguros establecidos. El proyecto se desarrollará aplicando principios de ingeniería electrónica y programación, evaluando el desempeño del prototipo en condiciones controladas antes de su posible implementación en campo.

Objetivos:

- Mejorar la seguridad laboral y ambiental en los criaderos de pollos mediante la detección temprana de gases tóxicos.
- Prevenir enfermedades y daños en la salud de los trabajadores y animales.
- Optimizar las condiciones ambientales dentro de las instalaciones productivas.
- Validar la precisión y efectividad del sistema en la medición y respuesta ante concentraciones peligrosas de gases.

Suposiciones y restricciones:

Como Restricciones tenemos utilizar Raspberry pi 4 y un set de sensores GrovePi+, sensores especializados para detectar gases tales como:

- CO (monóxido de carbono)
- CO2 (Dióxido de carbono)
- CH4 (Metano)
- NH3 (Amoníaco)
- H2S (Sulfuro de Hidrógeno)

el proyecto debe estar completo para el 15 de diciembre de 2025

Entregables:

Maqueta 3D , informe Etapa 13

Historial de versiones :

3.1.-Organización del proyecto

3.2.-Personal y entidades internas:

El equipo de trabajo está conformado por **tres programadores, dos analistas, un jefe de proyecto y un documentador.**

Cada integrante cumple funciones específicas orientadas al desarrollo, control y documentación del sistema de monitoreo de gases en los criaderos de pollos **Ariztía**, garantizando una correcta gestión del tiempo, calidad del software y validez técnica del prototipo.

3.3.-Roles y responsabilidades:

Programadores : Están a cargo de la **codificación del sistema**. Desarrollan el software que permite la lectura de los sensores (MQ-4, MQ-7, MQ-135) a través del convertidor ADC MCP3008 y la Raspberry Pi 4. Implementan las funciones de adquisición de datos, almacenamiento, alertas y visualización en pantalla o dashboard.

Analista :Son responsables de **investigar las condiciones ambientales del criadero**, definir los rangos de gases aceptables y establecer los requerimientos técnicos del sistema. Además, evalúan la viabilidad de los sensores, realizan pruebas de calibración y validan los resultados obtenidos.

Jefe de proyecto: Es el encargado de **dirigir y supervisar todas las etapas del desarrollo**, asegurando que las tareas se cumplan dentro de los plazos y con la calidad establecida. Coordina al equipo de trabajo, gestiona los recursos, controla los riesgos y verifica que el prototipo del sistema funcione correctamente para cumplir los objetivos definidos por el proyecto.

Documentador: Tiene la responsabilidad de **elaborar y mantener toda la documentación del proyecto**, incluyendo bitácoras de trabajo, planillas de seguimiento, actas de reuniones, manuales de usuario, informes de avance y reportes técnicos finales. Garantiza la trazabilidad y claridad de toda la información generada durante el proyecto.

3.4.-Mecanismos de Comunicación:

El equipo se comunica mediante **reuniones semanales** para coordinar avances y resolver problemas técnicos.

Se utiliza un **grupo de WhatsApp** para la comunicación diaria y rápida entre los integrantes.

Los documentos, bitácoras e informes se almacenan en **Google Drive**, mientras que el **código fuente** se gestiona en **GitHub**.

Las entregas formales y validaciones con el profesor se realizan por **correo institucional**.

4.-Planificación de los procesos de gestión

4.1.-Planificación inicial del proyecto:

Planificación de estimaciones:

Tomando en cuenta que se han utilizado seis semanas para la búsqueda de una problemática en la región y para el análisis de los objetivos más urgentes, se ha seleccionado una solución informática, completando así la primera fase del proyecto. Esto da paso a la siguiente etapa: su ejecución.

Para la ejecución, el horario de clases destinado al desarrollo será de 4 horas semanales, lo que nos deja un total de 7 semanas para completar esta fase, equivalente a 28 horas de trabajo. Además, se realizarán reuniones externas todos los viernes, con una duración de 1 hora, destinadas a coordinar, analizar y decidir sobre el rumbo del proyecto.

En total, se proyectan 35 horas de trabajo hasta la fecha de entrega y 65 horas en total.

Costo del software de desarrollo: \$0 (software libre).

Tiempo para programación: 2 meses

Integrantes	rol	Sueldo/hora	Total
Kamir Alfaro	Analista Programador Documetnador	\$15.000	\$975.000
Edson Galdames	Analista Programador Documetnador	\$15.000	\$975.000
Lukas Torres	Jefe de Proyecto Programador	\$17.000	\$1.105.000
Total			\$3.055.000

Costo sensores y convertidor analogico-sensorial

Sensores	Gas detectado	Precio estimado clp
MQ-4	CH4(metano)	\$1.115
MQ-7	CO(monóxido de carbono)	\$1.096
MQ-135	NH3 y CO2 (Amoniaco y Dióxido de carbono)	\$1.193
ADC	Convertidor analogico-digit al	\$3490
Kit Raspberry pi 4 8GB	MiniComputador	\$117.000

Planificación de recursos humanos :

Programadores : Kamir Alfaro, Edson Galdames y Lukas Torres.

Jefe de proyecto: Lukas Torres.

Analistas : Kamir Alfaro , Edson Galdames.

Documentador: Kamir Alfaro, Edson Galdames.

4.2.-Lista de Actividades:

Actividades de trabajo:

Planificación del Proyecto

Durante la etapa de planificación, desarrollada entre el 9 de septiembre y el 27 de octubre, se estableció el panorama general del proyecto, junto con la organización del equipo y la definición de objetivos.

El grupo determinó como propósito principal el diseño y desarrollo de un sistema detector de gases nocivos para una granja de gallinas, utilizando sensores MQ-4, MQ-7 y MQ-135 conectados a una Raspberry Pi. Este sistema busca monitorear la calidad del aire y emitir alertas en caso de concentraciones peligrosas de gases como metano (CH_4), Hidrógeno de Sulfuro (H_2S), monóxido de carbono (CO) o amoníaco (NH_3).

En esta fase, también se distribuyeron los roles de trabajo entre los integrantes, considerando tareas de investigación, programación, modelado 3D y documentación técnica.

Asimismo, se realizó la capacitación en el uso de las gafas Meta Quest (16 de septiembre) para la futura visualización inmersiva del sistema, y se elaboró la maqueta 3D del entorno entre el 16 de septiembre y el 14 de octubre, representando la instalación del sistema en una granja avícola.

Finalmente, se confeccionó el Informe de Etapa 1 (14 al 27 de octubre), que integró los avances y la planificación detallada del desarrollo posterior.

Ejecución del Proyecto

La ejecución del proyecto se llevará a cabo entre el 28 de octubre y el 14 de diciembre, diseño, codificación e integración.

Durante el análisis, se identificaron los gases críticos para el entorno avícola y se definieron los umbrales de detección adecuados para proteger la salud de las aves y mantener las condiciones óptimas del recinto.

En la fase de diseño, se establecerá la arquitectura del sistema, integrando los sensores MQ-4, MQ-7 y MQ-135 con la Raspberry Pi mediante protocolos de lectura analógica y digital. Se planificará la interfaz de visualización y la lógica de procesamiento de datos. Posteriormente, en la etapa de codificación, se desarrollarán los scripts para la lectura de datos de los sensores, almacenamiento de mediciones y generación de alertas.

Cierre del Proyecto

El cierre del proyecto está programado para el 14 de diciembre, fecha en la que se entregará el informe final y la demostración funcional del sistema.

Durante esta etapa se realizará la revisión de los resultados, la evaluación del cumplimiento de objetivos y la firma de las actas de conformidad por parte del equipo. Esta fase marca la conclusión formal del proyecto, consolidando los aprendizajes técnicos, la colaboración grupal y la aplicación práctica de tecnologías de detección y automatización en entornos reales.

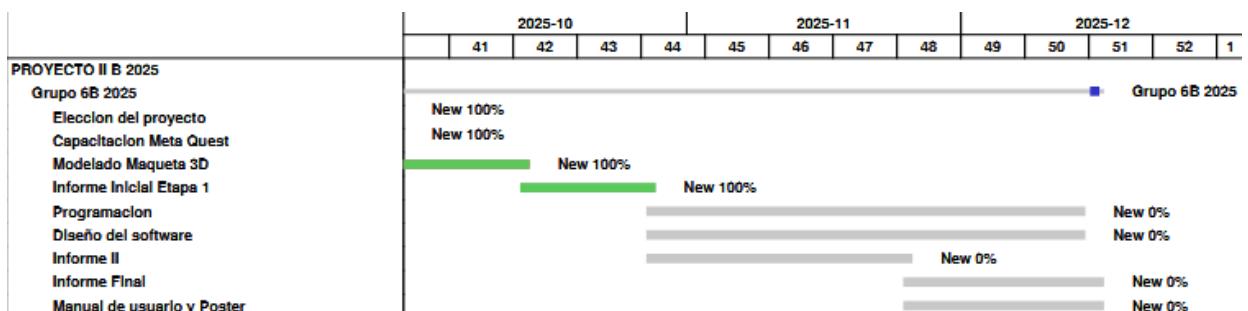
Planificación del proyecto:

A continuación, se presenta la planificación general del proyecto, organizada por etapas, indicando las fechas y objetivos principales de cada una:

Etapa	Periodo	Descripción de actividades
Elección del proyecto	9 de septiembre - 13 de septiembre	Durante esta etapa se definió el tema central del proyecto, su objetivo general y los recursos necesarios para su ejecución. Además, se realizó la distribución de roles y responsabilidades dentro del equipo de trabajo.
Capacitación Meta Quest	16 de septiembre	Se llevó a cabo una jornada de capacitación sobre el uso de las gafas Meta Quest, con el fin de adquirir las competencias técnicas necesarias para el desarrollo y visualización del entorno 3D.
Diseño de maqueta 3D	16 de septiembre - 14 de octubre	En esta fase se elaboró la maqueta tridimensional del proyecto, utilizando herramientas digitales de modelado 3D. Se buscó representar de manera precisa la estructura, componentes y funcionalidad del sistema propuesto.
Informe Etapa 1	14 de octubre - 27 de octubre	Se elaboró el primer informe técnico, que incluyó el diagnóstico, objetivos, planificación y avances iniciales del diseño.
Programación del proyecto	28 de octubre - 14 de diciembre	Etapa dedicada a la implementación del proyecto, integrando los componentes físicos y digitales. Incluye pruebas, ajustes y mejoras al sistema.
Diseño final y documentación	28 de octubre - 14 de diciembre	En paralelo con el desarrollo, se completó el diseño definitivo y la documentación técnica del proyecto, asegurando su coherencia con los objetivos planteados.
Informe II	28 de octubre - 25 de noviembre	Elaborar un informe intermedio que detalle los avances obtenidos, el montaje de los sensores, las primeras lecturas y las pruebas iniciales del sistema. Debe incluir resultados preliminares y observaciones sobre posibles mejoras.

Informe Final	25 de noviembre - 16 de diciembre	Desarrollar el documento final del proyecto, integrando todos los resultados, análisis y conclusiones. Incluir la validación de las lecturas de los sensores, la interpretación de los datos y la evaluación general del desempeño del sistema.
Manual de usuario y Poster	25 de noviembre - 16 de diciembre	Crear un manual con las instrucciones de uso, mantenimiento y calibración del sistema, junto con un póster explicativo que resuma los objetivos, metodología, resultados y conclusiones del proyecto para su presentación final.

Asignación de tiempo:



Elección de proyecto: 9 de Septiembre hasta 13 de Septiembre.

Capacitación Meta Quest: 16 de Septiembre.

Modelado maqueta 3D: 16 de Septiembre hasta 14 de Octubre.

Informe Inicial etapa 1: 14 de octubre hasta 28 de Octubre

Programación: 28 de Octubre hasta 14 de Diciembre

Diseño de software: 28 de Octubre hasta 14 de Diciembre

Informe II: 28 de Octubre hasta 25 de Noviembre

Informe Final: 25 de Noviembre hasta 16 de Diciembre

Manual de usuario y Póster: 25 de Noviembre hasta 16 de Diciembre

4.3.-Planificación de la gestión de riesgos:

Riesgo	Probabilidad ocurrencia	Nivel impacto	Acción remedial
Fallo en sensores MQ por humedad o temperatura extrema	70%	3 (Crítico)	Proteger sensores con cubiertas plásticas perforadas y filtros; mantener ventilación controlada.
Error en la calibración MQ-135 (NH3) o lecturas inestables	60%	3 (Crítico)	Realizar pruebas previas con valores conocidos; recalibrar periódicamente en aire limpio.
Sobrecarga eléctrica o falla en fuente de alimentación de la Raspberry Pi	40%	2 (Marginal)	Usar fuente de 5 V 3 A certificada y fusible de protección; disponer de fuente secundaria.
Fallo de conexión entre sensores y ADC MCP3008	50%	2 (Marginal)	Verificar cables y pines SPI; usar cables cortos y revisar soldaduras.
Deterioro de componentes por exposición prolongada a gases	30%	2 (Marginal)	Planificar reemplazo cada 6 meses; limpiar los módulos tras cada prueba.
Fallo de software o bloqueo script de lectura	40%	2 (Marginal)	Implementar watchdog o reinicio automático del proceso Python.
Error en el registro o pérdida de datos de medición	35%	2 (Marginal)	Implementar respaldo automático de datos (CSV o base SQLite).
Falta de experiencia integración con Raspberry Pi	60%	3 (Crítico)	Capacitación previa del equipo y uso de ejemplos funcionales.

Tipo de riesgo	Indicadores potenciales
Tecnológico	Lecturas erróneas, fallas en módulos MQ o ADC, inestabilidad en el software.
Humano/ Equipo	Desconocimiento técnico, errores en montaje o calibración.
Organizacional	Falta de coordinación entre integrantes, retrasos en tareas.
Entorno	Exposición a gases, polvo o humedad que afecten al hardware.
Estimación/ Tiempo	Subestimación del tiempo necesario para montaje y pruebas.

Avances del Proyecto

Durante la Etapa II se completó el montaje físico del sistema de medición ambiental, instalando los sensores MQ-4, MQ-7 y MQ-135 en conjunto con la Raspberry Pi y el convertidor analógico-digital para la lectura de datos. Se realizaron las primeras pruebas funcionales, validando la comunicación entre los sensores y el sistema de adquisición.

Las primeras lecturas registradas fueron:

- MQ-4 (metano): 0 ppm
- MQ-7 (monóxido de carbono): 67 ppm
- MQ-135 (contaminantes del aire / amoníaco): 0 ppm

Estas mediciones confirman el correcto funcionamiento del sistema de lectura, permitiendo continuar con la fase de calibración y análisis comparativo de valores.

Evidencia Fotográfica

Figura 1. Montaje físico del sistema.

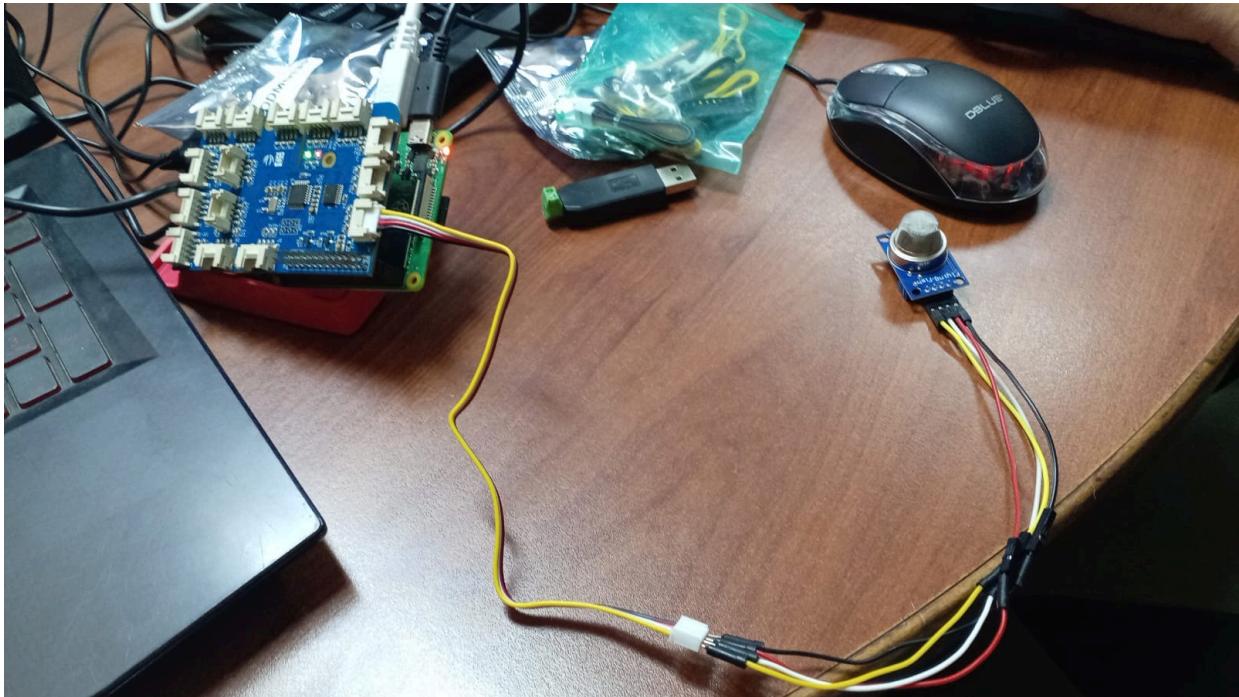


Figura 2. Conexión de sensores MQ a la Raspberry Pi y ADC.

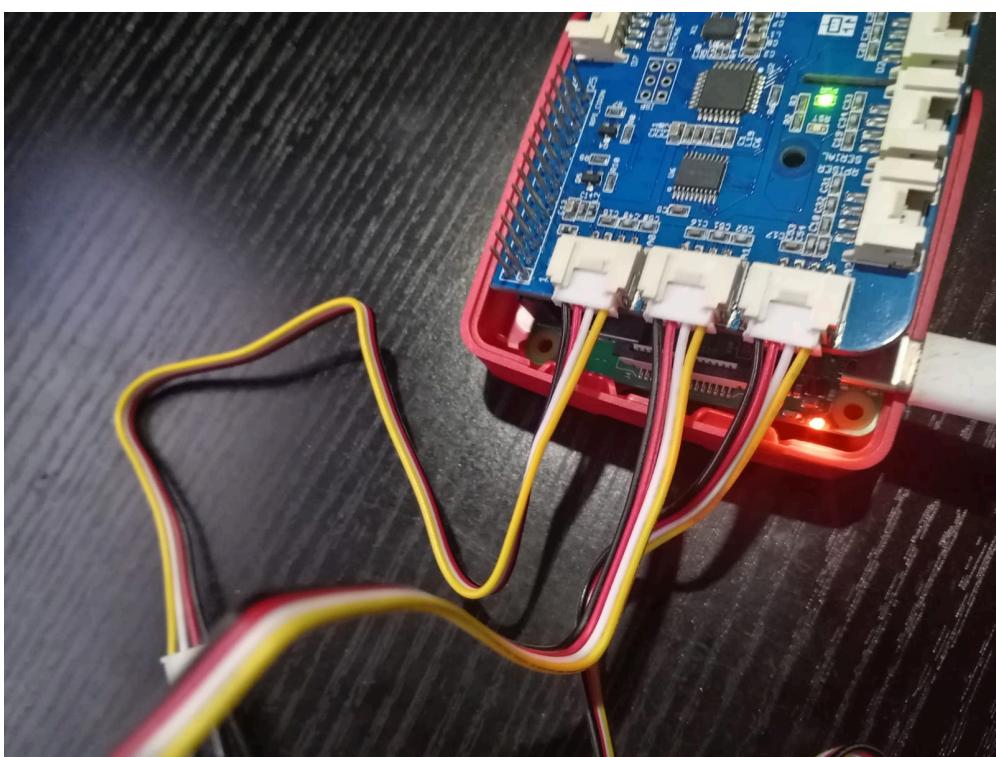
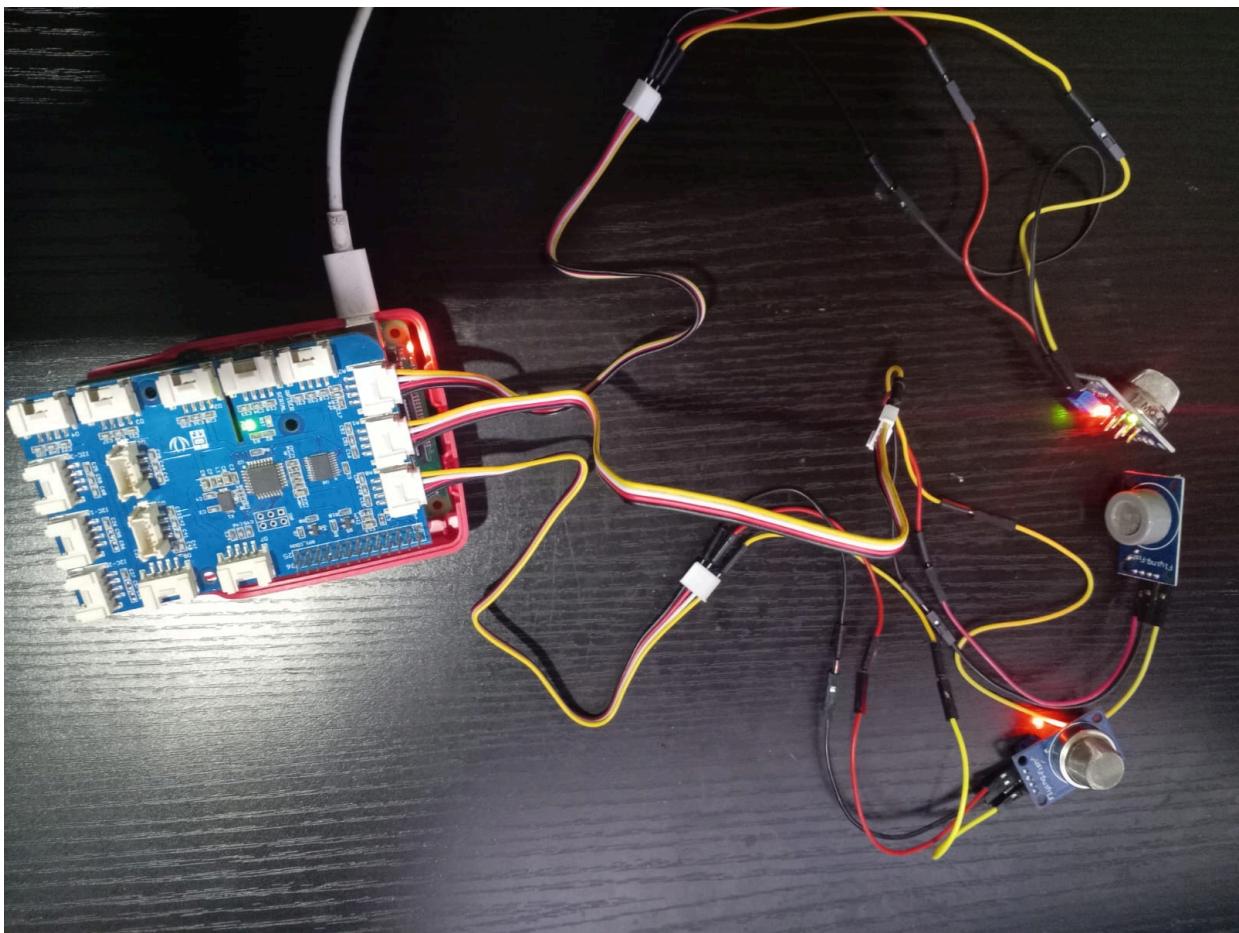


Figura 3. Primeras lecturas obtenidas del sistema.

```
Iniciando lectura del sensor MQ-7 (Monóxido de Carbono)...
Calentamiento: Los sensores MQ necesitan un tiempo de calentamiento (varios minutos) antes de dar lecturas estables.
Puerto A0: Valor Analógico Crudo = 67
Puerto A0: Valor Analógico Crudo = 62
Puerto A0: Valor Analógico Crudo = 61
Puerto A0: Valor Analógico Crudo = 59
Puerto A0: Valor Analógico Crudo = 135
Puerto A0: Valor Analógico Crudo = 102
Puerto A0: Valor Analógico Crudo = 124
Puerto A0: Valor Analógico Crudo = 121
Puerto A0: Valor Analógico Crudo = 74
Puerto A0: Valor Analógico Crudo = 68
Puerto A0: Valor Analógico Crudo = 62
```

Figura 4. Segunda lectura con todos los sensores funcionando

```
MQ-135 (CO2/Aire): 135 (CO): 0.53 PPM
MQ-4 (CH4): 43.10 PPM
MQ-135 (CO2/Aire): 0.00 PPM
MQ-7 (CO): 0.59 PPM
MQ-4 (CH4): 42.69 PPM
MQ-135 (CO2/Aire): 0.00 PPM
MQ-7 (CO): 0.49 PPM
MQ-4 (CH4): 41.89 PPM
MQ-135 (CO2/Aire): 0.00 PPM
MQ-7 (CO): 0.48 PPM
MQ-4 (CH4): 41.23 PPM
MQ-135 (CO2/Aire): 0.00 PPM
MQ-7 (CO): 0.47 PPM
MQ-4 (CH4): 42.29 PPM
MQ-135 (CO2/Aire): 0.00 PPM
MQ-7 (CO): 0.46 PPM
MQ-4 (CH4): 153.11 PPM
MQ-135 (CO2/Aire): 0.00 PPM
MQ-7 (CO): 0.45 PPM
MQ-4 (CH4): 67.75 PPM
MQ-135 (CO2/Aire): 7.51 PPM
```

Plan de Integración

Planificación de los Procesos Técnicos

Los procesos técnicos contemplan las actividades relacionadas con el diseño y desarrollo del sistema de medición ambiental:

- Integración de los sensores MQ-4, MQ-7, MQ-135 con la Raspberry Pi.
- Implementación del convertidor analógico-digital para la lectura de valores eléctricos.
- Desarrollo del software de adquisición de datos en Python.
- Configuración de la interfaz gráfica para monitoreo remoto en red local.
- Validación de primeras lecturas y comprobación del funcionamiento del hardware.

Durante la siguiente etapa se continuará con:

- Calibración individual de sensores.
- Ajuste de parámetros para la detección de gases críticos.
- Implementación de sistema de alertas visuales y/o sonoras.
- Implementación de interfaz gráfica

Planificación de los Procesos de Soporte

Los procesos de soporte consideran actividades complementarias necesarias para mantener la continuidad del proyecto:

- Gestión documental de avances, actas y bitácoras.
- Coordinación de reuniones para control de progreso.
- Control de inventario y materiales del laboratorio.
- Revisión de riesgos y medidas preventivas para resguardar los componentes electrónicos.

Estas acciones permiten asegurar organización, trazabilidad y resguardo del desarrollo del sistema.

Especificaciones de Requerimientos

La especificación de requerimientos define las funciones, condiciones y características que debe cumplir el sistema de detección de gases nocivos en plantas de Ariztía. Estos requerimientos permiten determinar con claridad el alcance técnico del proyecto y asegurar que el prototipo desarrollado satisfaga las necesidades planteadas en su diseño.

A continuación, se clasifican los requerimientos del sistema en dos categorías: funcionales y no funcionales.

Requerimientos Funcionales

- RF1: El sistema debe detectar la presencia de metano (CH_4) mediante el sensor MQ-4.
- RF2: El sistema debe detectar monóxido de carbono (CO) mediante el sensor MQ-7.
- RF3: El sistema debe detectar contaminantes del aire y amoníaco mediante el sensor MQ-135.
- RF4: El sistema debe almacenar las lecturas de los sensores en tiempo real.
- RF5: El sistema debe generar alertas cuando las concentraciones de gases superen los umbrales establecidos.
- RF6: El sistema debe mostrar gráficamente los valores obtenidos en una interfaz de monitoreo.
- RF7: El sistema debe permitir la visualización remota dentro de la red local.
- RF8: El sistema debe actualizar los datos en intervalos regulares sin necesidad de reiniciar el programa.
- RF9: El sistema debe permitir configurar los parámetros para emitir alerta.

Requerimientos no Funcionales

- RNF1: El sistema debe operar continuamente sin interrupciones durante toda la jornada de funcionamiento del plantel.
- RNF2: La interfaz de usuario debe ser clara, legible y comprensible para usuarios sin conocimientos técnicos.
- RNF3: El sistema debe tener un tiempo de respuesta menor a 3 segundos entre cada actualización de datos.
- RNF4: El sistema debe almacenar la información de forma segura y sin pérdida de datos.
- RNF5: El sistema debe funcionar sin poner en riesgo la integridad del hardware electrónico.
- RNF6: El software debe ser portable y ejecutarse correctamente en cualquier equipo con Python y PyQt5.
- RNF7: La comunicación entre la Raspberry y la interfaz remota debe garantizar estabilidad dentro de la red local.
- RNF8: El código debe estar versionado y respaldado mediante el repositorio de GitHub del proyecto.

Criterios de Aceptación

CA1: El sistema será considerado operativo si los tres sensores MQ entregan valores correctamente en tiempo real.

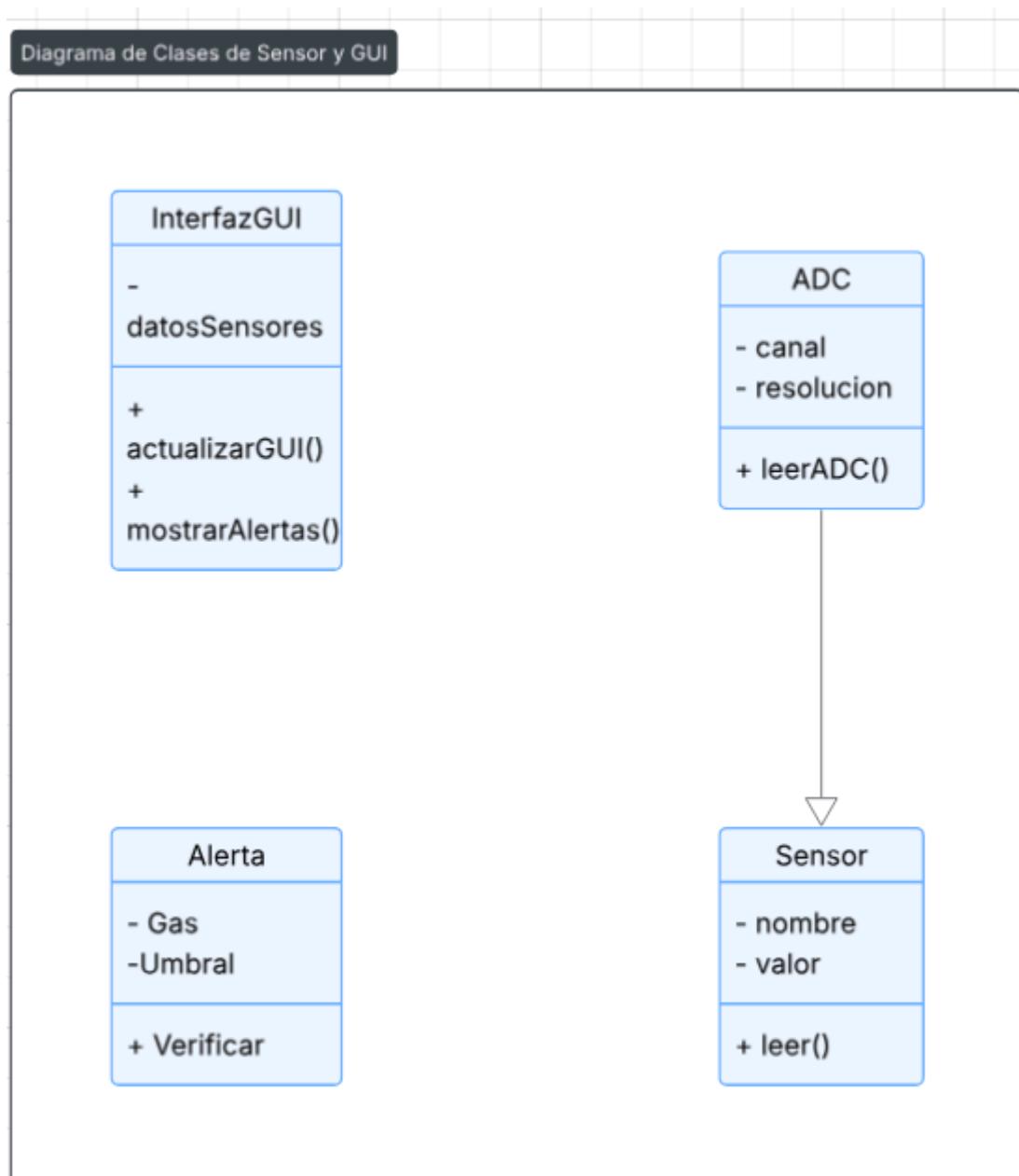
CA2: El sistema será considerado seguro si las alarmas se activan al superar los umbrales establecidos.

CA3: El sistema será considerado estable si puede operar durante 6 horas continuas sin reinicios ni bloqueos.

CA4: El sistema será evaluado como completo cuando la interfaz gráfica permita monitoreo remoto en la red local.

Diagrama de Clases

El diagrama de clases representa la estructura principal del software del sistema de detección de gases nocivos. Se modelaron las clases esenciales que intervienen en la adquisición de datos, el procesamiento y la visualización de la información mediante la interfaz gráfica. El diseño se desarrolló bajo el paradigma de Programación Orientada a Objetos, facilitando la modularidad, la escalabilidad y el mantenimiento del proyecto.



El diagrama de clases presentado modela la estructura lógica del sistema de monitoreo de gases implementado mediante sensores conectados a un ADC, con visualización a través de una interfaz gráfica. Cada clase representa un componente clave del funcionamiento del sistema y define sus atributos y métodos principales.

Clase ADC

Representa el módulo de conversión analógica-digital encargado de transformar las señales analógicas provenientes de los sensores en valores digitales procesables.

Atributos:

- canal: número del canal asignado al sensor.
- resolución: resolución del ADC utilizada para la conversión.
Método:
 - leerADC(): obtiene el valor analógico convertido desde el canal correspondiente.

Clase Sensor

Modela cada sensor físico del sistema (MQ-4, MQ-7, MQ-135).

Atributos:

- nombre: identifica el tipo de sensor o el gas que detecta.
- valor: almacena la lectura obtenida del ADC.
Método:
 - leer(): solicita una lectura al ADC y actualiza el atributo valor.

La relación entre ADC y Sensor indica que un sensor utiliza al módulo ADC para poder obtener su medición.

Clase Alerta

Controla el sistema de alarmas basado en concentraciones peligrosas.

Atributos:

- Gas: identifica el gas asociado a la alerta.
- Umbral: nivel máximo permitido antes de activar la alerta.
Método:
 - Verificar(): compara el valor del sensor con el umbral y determina si se debe generar una advertencia.

Clase InterfazGUI

Corresponde a la ventana gráfica del software donde se visualizan los valores de los sensores y las alertas.

Atributos:

- datosSensores: conjunto de valores importados desde los sensores.

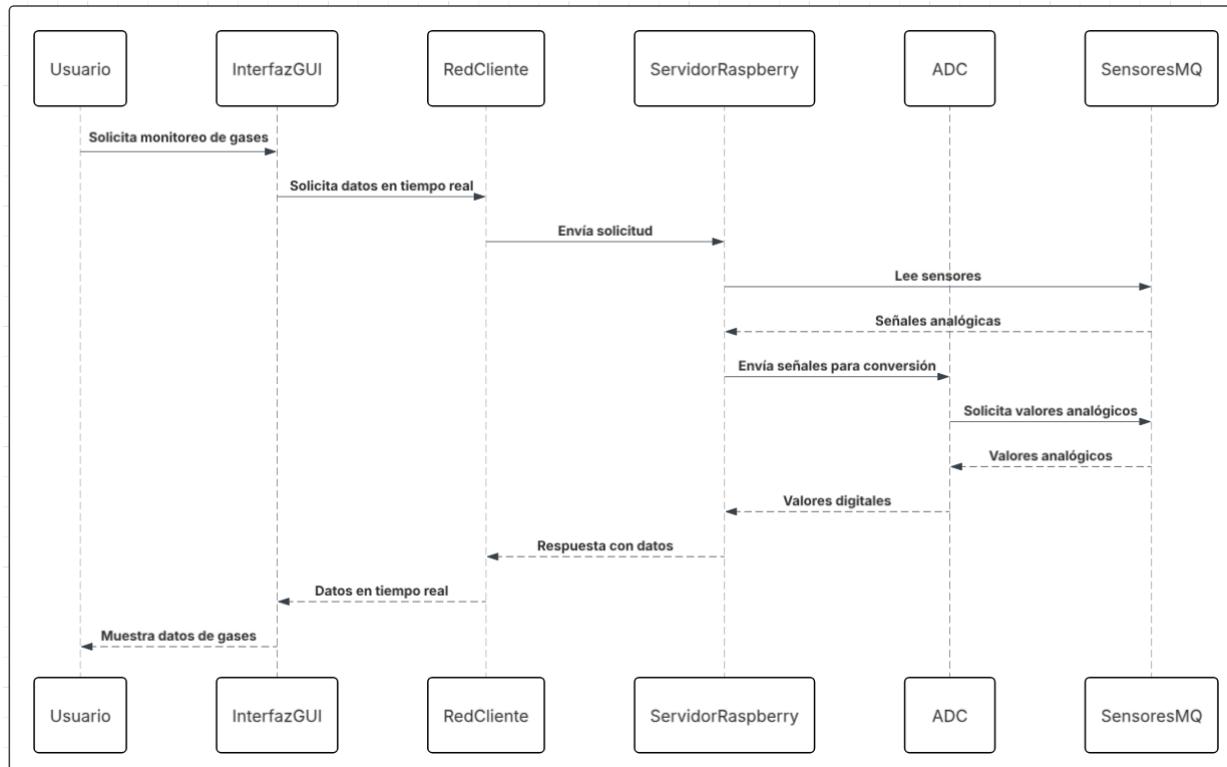
Métodos:

- actualizarGUI(): refresca la información en pantalla con los valores más recientes.

- mostrarAlertas(): despliega advertencias visuales si se detectan valores críticos.

Diagrama de Secuencia

El diagrama de secuencia representa el flujo dinámico de interacción entre los componentes del sistema durante una consulta de mediciones. Describe paso a paso la comunicación entre la Interfaz Gráfica, el módulo de conexión de red, la Raspberry Pi y los sensores, desde la solicitud de datos hasta la visualización final para el usuario.



El diagrama de secuencia modela el flujo de interacción temporal entre los actores y los componentes del sistema para llevar a cabo el monitoreo en tiempo real de los gases ambientales. Representa paso a paso cómo se envían y reciben los mensajes desde que el usuario solicita información hasta que los datos son procesados y visualizados en la interfaz gráfica.

Secuencia general del proceso

1. Usuario → InterfazGUI

El usuario inicia la acción solicitando visualizar el monitoreo de gases.

2. InterfazGUI → RedCliente

La interfaz gráfica envía una solicitud de datos a través de la red para obtener información en tiempo real.

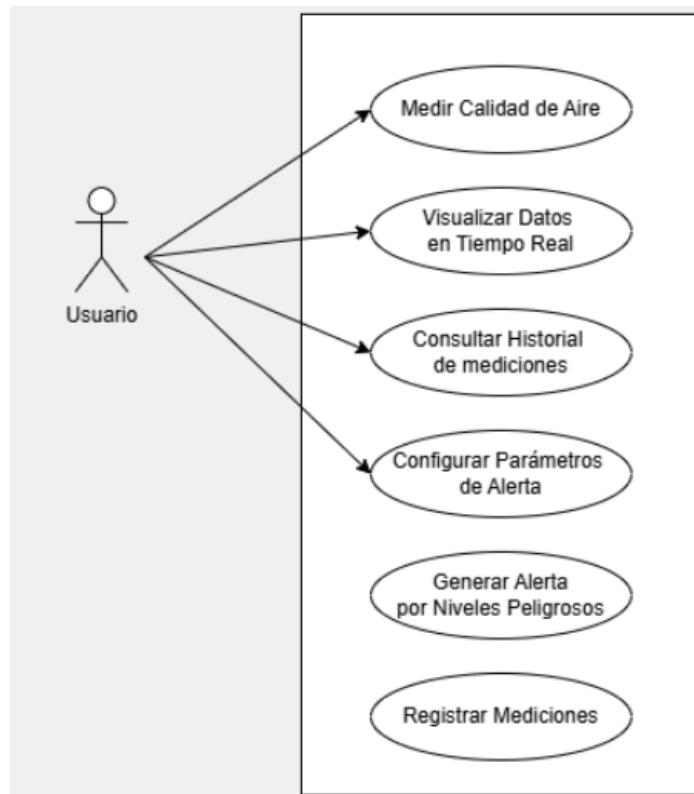
3. RedCliente → ServidorRaspberry

La solicitud llega al servidor alojado en la Raspberry Pi, donde se ejecuta el software encargado de la captura de datos.

4. **ServidorRaspberry → SensoresMQ**
El servidor solicita lecturas a los sensores MQ (MQ-4, MQ-7, MQ-135).
5. **SensoresMQ → ServidorRaspberry**
Los sensores entregan señales analógicas representando las concentraciones de gases.
6. **ServidorRaspberry → ADC**
El servidor envía las señales analógicas al ADC para su conversión.
7. **ADC → ServidorRaspberry**
El ADC procesa las señales y devuelve valores digitales correspondientes.
8. **ServidorRaspberry → RedCliente**
El servidor encapsula los datos digitales y envía una respuesta completa con los valores obtenidos.
9. **RedCliente → InterfazGUI**
La respuesta es transmitida a la aplicación GUI para que los valores puedan ser visualizados.
10. **InterfazGUI → Usuario**
La interfaz actualiza la pantalla y muestra los datos de concentración de gases en tiempo real.

Diagrama Caso de uso General

El diagrama de casos de uso representa, de forma conceptual, las funcionalidades principales que ofrece el sistema de monitoreo de gases nocivos al usuario final. El actor identificado es el **Usuario**, quien interactúa directamente con la interfaz del sistema para supervisar el ambiente de los criaderos avícolas y tomar decisiones de seguridad cuando sea necesario.



Medir Calidad de Aire: Permite iniciar el proceso de captación de datos a partir de los sensores MQ instalados en el recinto.

Visualizar Datos en Tiempo Real: Presenta en la interfaz gráfica las mediciones actuales tomadas por los sensores, permitiendo un monitoreo continuo.

Consultar Historial de Mediciones: Ofrece acceso a registros almacenados previamente para analizar tendencias u ocurrencias pasadas.

Configurar Parámetros de Alerta: Facilita la definición de límites críticos de concentración para cada gas, ajustándose a las condiciones de la instalación.

Generar Alerta por Niveles Peligrosos: Se activa automáticamente cuando se detectan valores que superan los umbrales establecidos, notificando al usuario sobre el riesgo.

Registrar Mediciones: Guarda las lecturas provenientes de los sensores.

Para revisar casos de uso a detalle referirse al otro documento subido al Redmine

Descripción de la Arquitectura del Sistema

La arquitectura del sistema se basa en un modelo distribuido cliente–servidor, en el cual la Raspberry Pi actúa como servidor encargado de separar las tareas de lectura y procesamiento físico de sensores, del análisis y monitoreo para la adquisición de datos ambientales y el procesamiento inicial de la información, mientras que un Smartphone funciona como cliente, responsable de la visualización remota de los resultados. Este diseño es puntual, optimizando el rendimiento y la seguridad del sistema.

El sistema se estructura en tres niveles funcionales:

1. Capa de Sensores (Modelo Físico-Electrónico)

Corresponde al nivel encargado de la obtención de las variables ambientales. Está conformado por los sensores MQ-4, MQ-7 y MQ-135, los cuales detectan gases nocivos y parámetros ambientales. Las señales producidas por los sensores son analógicas, por lo que la capa incorpora un convertidor analógico-digital (ADC), encargado de traducir estas señales para su lectura mediante la Raspberry Pi.

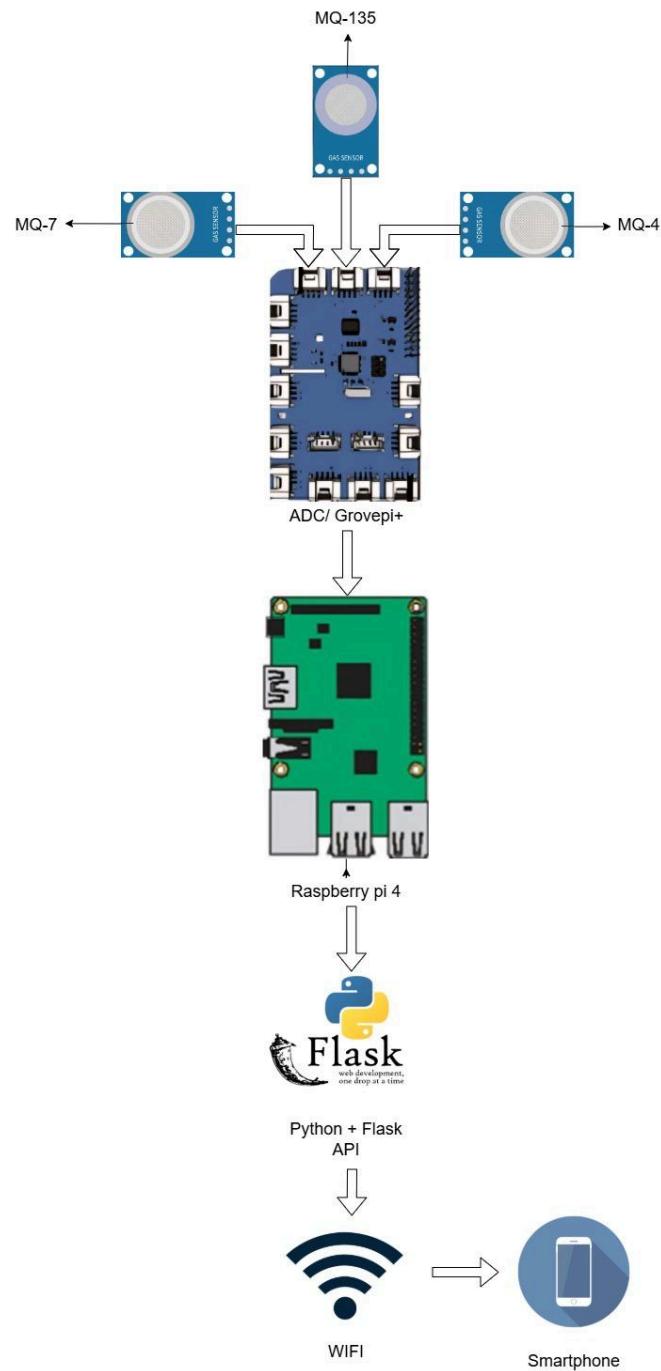
2. Capa de Procesamiento (Modelo de Control)

Este nivel se ejecuta dentro de la Raspberry Pi, donde el software en Python administra la adquisición periódica de las mediciones provenientes del ADC, almacena los datos y evalúa si se superan los umbrales establecidos. En esta capa se implementa un módulo de comunicaciones encargado de encapsular los valores en paquetes estructurados y transmitirlos por red local al cliente remoto.

3. Capa de Visualización (Modelo de Interfaz)

Corresponde a la estación del usuario remoto, donde se ejecuta la interfaz gráfica desarrollada con PyQt5. Esta capa recibe los datos enviados por la Raspberry Pi y actualiza en tiempo real los valores mostrados al usuario. La interfaz interpreta la información y activa alertas visuales o sonoras cuando se detectan riesgos, permitiendo monitoreo no presencial y a distancia.

Cada modelo opera de manera independiente, pero se comunica mediante protocolos estructurados, facilitando la modularidad, el mantenimiento y la posibilidad de futuras ampliaciones del sistema, como la incorporación de nuevos sensores o la integración de una base de datos histórica.



Conclusión Etapa II

Durante la Etapa II se lograron avances fundamentales en la construcción del sistema de monitoreo de gases nocivos para plantas avícolas. Se completó el montaje físico de los sensores MQ (MQ-4, MQ-7 y MQ-135) junto con su conexión al ADC y a la Raspberry Pi, permitiendo obtener las primeras lecturas reales de concentración de gases presentes en el ambiente, específicamente metano (CH_4), monóxido de carbono (CO), amoníaco (NH_3) y sulfuro de hidrógeno (H_2S). Estas pruebas iniciales confirmaron la correcta comunicación entre los sensores y la Raspberry, validando la factibilidad técnica del sistema planteado.

De forma paralela al montaje electrónico, se avanzó en el desarrollo de la interfaz gráfica destinada al monitoreo remoto del sistema. Si bien esta etapa no contempló la finalización de la interfaz, se establecieron las bases de su estructura, conexión con el sistema de captura de datos y diseño preliminar para la visualización en tiempo real. El sistema se enfocará exclusivamente en el monitoreo de gases, descartando la inclusión de temperatura y humedad para esta versión del prototipo, lo que permitió concentrar los esfuerzos en la detección más precisa de los compuestos nocivos.

En síntesis, la Etapa II se considera exitosa, ya que permitió validar el funcionamiento electrónico del sistema y progresar en la construcción del módulo visual, dejando al proyecto en una posición favorable para iniciar la etapa final. Los próximos pasos estarán centrados en completar la interfaz gráfica, implementar el módulo de alertas, optimizar la presentación de datos y mejorar la estabilidad del sistema para su futura demostración.

Referencias

- [MQ-4](#)
- [MQ-7](#)
- [MQ-135](#)
- [MICS-5524](#)
- [Información Avicola](#)

[Link de compra sensores](#)

[MQ-4, MQ-7, MQ-135](#)