

**UNIVERSIDAD PRIVADA DE TACNA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA
ELECTRÓNICA**



TESIS

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO IOT PARA
LA MEDICIÓN Y MONITOREO DE LA CALIDAD DE AIRE EN
LAS SALAS DE FUMADORES DE UN CASINO”**

PARA OPTAR:

TÍTULO PROFESIONAL DE INGENIERO ELECTRÓNICO

PRESENTADO POR:

Bach. RICHARD ANDRÉS CUADROS TAMAYO

Bach. JORGE LUIS CHUCULLA HUAYHUA

TACNA – PERÚ

2025

**UNIVERSIDAD PRIVADA DE TACNA FACULTAD DE
INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA**

TESIS

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO IOT PARA
LA MEDICIÓN Y MONITOREO DE LA CALIDAD DE AIRE EN
LAS SALAS DE FUMADORES DE UN CASINO”**

Tesis sustentada y aprobada el 10 de diciembre del 2025; estando el jurado calificador integrado por:

PRESIDENTE : Mtro. MARIA ELENA VILDOZO ZAMBRANO

SECRETARIO : Mtro. CARLOS ARMANDO RODRIGUEZ SILVA

VOCAL : M Sc. MILAGROS GLENY COHAILA GONZALES

ASESOR : Dr. ANIBAL JUAN ESPINOZA ARANCIAGA

DECLARACIÓN JURADA DE ORIGINALIDAD

Nosotros, Richard Andrés Cuadros Tamayo y Jorge Luis Chuculla Huayhua, bachilleres, de la Escuela Profesional de Ingeniería Electrónica de la Facultad de Ingeniería de la Universidad Privada de Tacna, identificados con DNI N° 44812311 y N°45870212 respectivamente, así como Aníbal Juan Espinoza Aranciaga con DNI N° 06154874, declaramos en calidad de autores y asesor que:

1. Somos autores de la tesis titulada: "*Diseño e implementación de un prototipo iot para la medición y monitoreo de la calidad de aire en las salas de fumadores de un casino*", la cual presentamos para optar el Título Profesional de Ingeniero Electrónico.
2. La tesis es completamente original y no ha sido objeto de plagio, total ni parcialmente, habiéndose respetado rigurosamente las normas de citación y referencias para todas las fuentes consultadas.
3. Los datos presentados en los resultados son auténticos y no han sido objeto de manipulación, duplicación ni copia.

En virtud de lo expuesto, asumimos frente a La Universidad toda responsabilidad que pudiera derivarse de la autoría, originalidad y veracidad del contenido de la tesis, así como por los derechos asociados a la obra.

En consecuencia, nos comprometemos ante a La Universidad y terceros a asumir cualquier perjuicio que pueda surgir como resultado del incumplimiento de lo aquí declarado, o que pudiera ser atribuido al contenido de la tesis, incluyendo cualquier obligación económica que debiera ser satisfecha a favor de terceros debido a acciones legales, reclamos o disputas resultantes del incumplimiento de esta declaración.

En caso de descubrirse fraude, piratería, plagio, falsificación o la existencia de una publicación previa de la obra, aceptamos todas las consecuencias y sanciones que puedan derivarse de nuestras acciones, acatando plenamente la normatividad vigente.

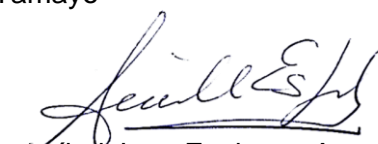
Tacna, 19 de diciembre del 2025



Richard Andrés Cuadros Tamayo
DNI:44812311



Jorge Luis Chuculla Huayhua
DNI: 45870212



Aníbal Juan Espinoza Aranciaga
DNI: 06154874

DEDICATORIA

El Presente trabajo se lo dedicamos con mucho empeño y labor realizada a nuestras familias, que han sido el apoyo incondicional para la elaboración del presente proyecto, el agradecimiento a dios ya que de su mano hemos podido completar las metas propuestas, el agradecimiento a los docentes que fueron parte del proceso ya que nos brindaron tiempo y conocimiento para nuestro desarrollo profesional y las herramientas suficientes durante nuestro proceso académico, a nuestros amigos que compartieron aulas con nosotros durante la formación académica y a su vez esperando puedan completar las metas y sueños propuestos.

Richard Andrés Cuadros Tamayo

Jorge Luis Chuculla Huayhua

AGRADECIMIENTO

El agradecimiento a los docentes de la escuela profesional de Ingeniería Electrónica, por apoyarnos en cada problema presentado durante la realización de la tesis presente y por darnos el conocimiento adecuado para culminarla.

Richard Andrés Cuadros Tamayo

Jorge Luis Chuculla Huayhua

Agradezco a Dios, por su presencia a lo largo de mi vida, por las enseñanzas y oportunidades, pero sobre todo por la maravillosa familia que me acompaña y no me abandona ya que gracias a su amor incondicional me dan la motivación y fuerza para continuar con mis proyectos y de esta manera poder sumar grandes logros, siendo la culminación de este uno de los más importantes en mi vida y la de mi familia. Con amor para mis padres.

Richard Andrés Cuadros Tamayo

Agradezco a Dios, por todo el apoyo emocional y en espíritu del cual me he sentido apto y capaz de culminar la tesis presente, a mi familia por el apoyo incondicional que he recibido a diario durante esta etapa de mi vida, a pesar de cada adversidad mis padres están y estarán presentes por siempre.

Una dedicatoria a mi hijo Josué, que por el hago todo el esfuerzo que día a día se ve reflejado en mi trabajo, y espero que en un futuro no muy lejano yo pueda darte esa alegría que todo papá merece dar a sus hijos.

Jorge Luis Chuculla Huayhua

ÍNDICE GENERAL

PÁGINA DE JURADOS	ii
DECLARACIÓN JURADA DE ORIGINALIDAD	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE GENERAL.....	vi
ÍNDICE DE TABLAS	viii
ÍNDICE DE FIGURAS.....	ix
RESUMEN.....	xi
ABSTRACT.....	xii
INTRODUCCIÓN	1
CAPÍTULO I: EL PROBLEMA DE INVESTIGACIÓN.....	2
1.1. Descripción del problema	2
1.2. Formulación del problema	2
1.3. Justificación e Importancia	3
1.4. Objetivos	4
1.4.2. Objetivos específicos.....	4
1.5. Hipótesis	4
CAPÍTULO II: MARCO TEÓRICO.....	5
2.1. Antecedentes de la investigación	5
2.1.1. Antecedentes internacionales	5
2.1.2. Antecedentes nacionales.....	6
2.2. Bases teóricas.....	8
2.2.2. Internet de las cosas (IoT).....	9
2.2.2.1. Componentes específicos del internet de las cosas (IoT).....	10
2.2.3. Topología de red IoT	11
2.2.4. Medios de transmisión.....	13
2.2.5. Recopilación de datos en redes IoT.....	18
2.2.6. Normativas sobre la calidad del aire	19
2.3. Definición de términos	23
2.3.1. Calidad del aire.....	24
CAPÍTULO III: MARCO METODOLÓGICO.....	25
3.1. Diseño de la investigación.....	25
3.2. Acciones y actividades	25
3.2.1. Criterios iniciales.....	25

3.2.2. Selección de dispositivos	28
3.2.2.2. Micro-controlador	33
3.2.2.3. Interfaz gráfica	38
3.2.2.4. Medidor de calidad de aire	40
3.2.3. Desarrollo del prototipo	42
3.2.4. Desarrollo del software	48
3.3. Materiales e Instrumentos.	56
3.3.2. Instrumentos	57
3.4. Población y/o muestra de estudio.....	58
3.5. Operacionalización de variables.....	58
3.6. Procesamiento y análisis de datos.	59
3.6.2. Etapas del procesamiento.....	59
3.6.3. Diagrama de flujo del procesamiento de datos.....	60
CAPÍTULO IV: RESULTADOS.....	61
4.1. Medidor de calidad de aire	61
4.2. Prototipo IoT de medición de calidad de aire.....	61
4.3. Parámetros a medir de la calidad de aire	62
4.3.1. Monóxido de carbono (CO)	62
4.3.2. Dióxido de carbono (CO ₂)	63
4.3.3. Formaldehído (HCHO)	64
4.3.4. Material particulado 2.5 (PM _{2.5})	65
4.3.5. Temperatura (C°)	65
4.3.6. Humedad (Hr %)	66
CAPÍTULO V: DISCUSIONES	67
CONCLUSIONES	68
RECOMENDACIONES	69
REFERENCIAS BIBLIOGRÁFICAS	70
ANEXOS.....	73
Anexo 1: Matriz de consistencia.....	74
Anexo 2: Código de programación para el ESP32 en Arduino 1.8.15.....	75

ÍNDICE DE TABLAS

Tabla 1. Límites permitidos por la OMS.....	8
Tabla 2. Evolución del estándar 802.3xx respecto a la velocidad	17
Tabla 3. Estándar 802.11xx y su evolución respecto a la velocidad de transmisión ...	18
Tabla 4. Normas y Leyes relacionadas con la calidad del aire en Perú	23
Tabla 5. Contaminantes del aire en la sala de fumadores	24
Tabla 6. Cronograma de actividades	26
Tabla 7. Características técnicas de MH-Z19	30
Tabla 8. Características técnicas de PMS5003	31
Tabla 9. Características técnicas de DHT22 / AM2302	32
Tabla 10. Características técnicas de MS-1100 (CJM CU-1100).....	33
Tabla 11. Características técnicas de MICS-6814	34
Tabla 12. Características técnicas del ESP32	36
Tabla 13. Especificaciones técnicas del medidor de calidad de aire.....	43
Tabla 14. Consumo de corriente total del circuito.....	45
Tabla 15. Operacionalización de variables	61
Tabla 16. Lectura del equipo medidor de calidad de aire.....	64
Tabla 17. Lectura del prototipo IoT para las mediciones de calidad de aire.....	65
Tabla 18. Lecturas desde el medidor y prototipo IoT para el CO	66
Tabla 19. Lecturas desde el medidor y prototipo IoT para el CO ₂	66
Tabla 20. Lecturas desde el medidor y prototipo IoT para el formaldehído	67
Tabla 21. Lecturas desde el medidor y prototipo IoT para el PM 2.5	68

ÍNDICE DE FIGURAS

Figura 1. Medición del monóxido de carbono CO con un sensor MQ-7.....	6
Figura 2. Diagrama del sistema propuesto.....	6
Figura 3. Diagrama del sistema IOT basado en Arduino.....	7
Figura 4. Atributos de los Dispositivos IoT	10
Figura 5. Tipos de topología de redes.....	12
Figura 6. Tipos de medios guiados	13
Figura 7. Cable directo según TIA/EIA 568-A	14
Figura 8. Cable directo según TIA/EIA 568-B	15
Figura 9. Distribución de colores para cable cruzado tipo 1	15
Figura 10. Distribución de colores para cable cruzado tipo 2.....	16
Figura 11. Demostración del rango de alcance que tienen los enlaces de radio a un determinado estándar y frecuencia.....	17
Figura 12. Relación señal SNR y BER.....	18
Figura 13. Casino parte posterior acceso a sala de fumadores	26
Figura 14. Casino parte posterior de la sala de fumadores.....	26
Figura 15. Salas de fumadores denominadas sala 5 y sala 6	27
Figura 16. Casino plano en AUTOCAD	27
Figura 17. Topología de redes mesh	28
Figura 18. Sensor de CO2 modelo MHZ19.....	28
Figura 19. Sensor de material particulado PMS5003.....	29
Figura 20. Sensor de temperatura y humedad.....	30
Figura 21. Sensor de formaldehído modelo MS-1100.....	31
Figura 22. Microcontrolador ESP32	34
Figura 23. Diagrama de bloques del módulo ESP32.....	35
Figura 24. Área de cobertura para la aplicación del dispositivo IoT	37
Figura 25. Cobertura del módulo en el casino	37
Figura 26. Ubicación del AP y del prototipo IoT	37
Figura 27. Sala de control para el monitoreo de la calidad de aire.....	38
Figura 28. Utilidad de firebase en el monitoreo de calidad de aire en el casino	39
Figura 29. Plataforma web Firebase	40
Figura 30. Medidor de calidad de aire.....	42
Figura 31. Diagrama esquemático del circuito de la fuente de alimentación	45
Figura 32. Fuente modular Hi-Link HLK-PM05	45
Figura 33. Diagrama de conexión del módulo ESP32 de su fuente de alimentación.....	46

Figura 34. Esquemático del circuito de mando	47
Figura 35. Ubicación de componentes dentro del nodo final	47
Figura 36. Vista del entorno de IDE de Arduino.....	48
Figura 37. Configuración inicial.....	48
Figura 38. Función de interrupción de la señal PWM (> 1.0 um).....	49
Figura 39. Configuración de parámetros para la red malla.....	51
Figura 40. Código fuente	52
Figura 41. Código fuente	52
Figura 42. Topología final del proyecto.....	53
Figura 43. Estructura del desarrollo	54
Figura 44. Estructura general	54
Figura 45. Ubicación de la zona de despliegue de la red malla	55
Figura 46. Ubicación de la sala de fumadores	56
Figura 47. Vista de la ubicación del AP (Access point) dentro de la sala de fumadores.....	56
Figura 48. Diagrama de flujo de procesamiento de datos	60
Figura 49. Resultados de las 10 primeras muestras de cada sensor en tiempo real.....	66

RESUMEN

El presente proyecto de investigación tiene como objetivo el diseño e implementación de un prototipo IoT para la medición y monitoreo de la calidad de aire en las salas de fumadores de un casino. Debido a la alta concentración de contaminantes tóxicos generados por el humo de tabaco, se busca desarrollar un sistema integrado que permita la captura en tiempo real de parámetros ambientales críticos como partículas PM2.5, monóxido de carbono (CO), dióxido de carbono (CO2), temperatura y humedad. El prototipo IoT combinará sensores especializados con tecnología de comunicación inalámbrica para enviar los datos a una plataforma digital, facilitando la visualización, almacenamiento y generación de alertas tempranas. La implementación de este sistema permitirá optimizar la gestión ambiental del espacio cerrado, mitigando riesgos a la salud de usuarios y trabajadores, y promoviendo un ambiente más seguro. La validación del prototipo se realizará mediante pruebas experimentales en el entorno real. Este proyecto contribuye al desarrollo tecnológico en el área de control ambiental, proponiendo una solución innovadora, eficiente y accesible para la vigilancia continua de la calidad del aire en ambientes con presencia de humo de tabaco.

Palabras claves: Sistema IoT; Calidad del aire; Monitoreo y control en tiempo real.

ABSTRACT

This research project aims to design and implement an IoT prototype for measuring and monitoring air quality in the smoking rooms of a casino. Due to the high concentration of toxic pollutants generated by tobacco smoke, the objective is to develop an integrated system that captures in real time critical environmental parameters such as PM2.5 particles, carbon monoxide (CO), carbon dioxide (CO₂), temperature, and humidity. The IoT prototype will combine specialized sensors with wireless communication technology to transmit data to a digital platform, enabling visualization, storage, and early alert generation. Implementing this system will optimize environmental management in the enclosed space, mitigating health risks for users and workers, and promoting a safer environment. The prototype validation will be carried out through experimental tests in the real environment. This project contributes to technological advancement in environmental control, proposing an innovative, efficient, and accessible solution for continuous air quality surveillance in smoke-exposed indoor spaces.

Keywords: IoT system; Air quality; Real-time monitoring and control.

INTRODUCCIÓN

La presencia de contaminantes aéreos en ambientes cerrados, como en los casinos de juegos, es motivo de preocupación de la Organización Mundial de la Salud (OMS) debido a la afectación en la salud de los jugadores al quedar expuestos a partículas o gases que se emiten en dichos ambientes, lo que puede producir problemas respiratorios, cardíacos y afectar la función cognitiva. La combinación de un ambiente cerrado con deficiencias de ventilación y la presencia de una alta concentración de humo de tabaco y otros químicos provenientes de máquinas de juego, puede producir un impacto negativo en la salud de las personas que concurren a dichos locales, especialmente si tienen una ventilación deficiente.

Una de las soluciones tecnológicas en la actualidad, es la implementación de sistemas basados en el Internet de las Cosas (IoT) que permitan medir y monitorear en tiempo real la calidad del aire y optimizar las condiciones ambientales en las salas de fumadores para prevenir daños a la salud de los comensales y trabajadores en estos centros de entretenimiento.

La tesis se organizó de la siguiente: En el capítulo I se desarrolla la problemática existente en los casinos de juegos referido a los altos niveles de contaminantes en el aire de esos locales y se plantea una solución a través de implementar un prototipo de sistema IoT que mida los niveles de contaminantes en el aire. En el capítulo II se desarrolla la fundamentación teórica y se toma como antecedentes aquellas investigaciones nacionales e internacionales que puedan contribuir con el desarrollo de nuestra tesis. En el capítulo III se plantea la metodología de investigación empleada, describiendo las acciones y actividades a desarrollar, así como las técnicas, los instrumentos y el procesamiento de la información. En el capítulo IV se presenta los valores de las mediciones obtenidas con el prototipo desarrollado y con un medidor de calidad de aire. En el capítulo V se presenta un análisis comparativo de los valores obtenidos con el prototipo desarrollado y con un medidor de calidad de aire. Finalmente se termina con la presentación de las conclusiones a las que se llega y las recomendaciones a tomar en cuenta.

CAPÍTULO I: EL PROBLEMA DE INVESTIGACIÓN

1.1. Descripción del problema

Los juegos de azar se denotan desde las civilizaciones antiguas en donde se usan para entretenimiento de personas allegadas a resolver problemas financieros. Esto genera que se usen espacios cerrados que eleven los sus riesgos a nivel mundial debido a la calidad de aire, el fácil acceso a salas con esta tecnología lleve a la participación a niveles sin precedentes con consecuencias para la salud pública.

La expansión de la industria ha estado a la par junto al marketing que se le facilita la reincidencia de las personas a adquirir o reposar en las salas de juegos de los casinos más populares sin medir su propio malestar con los casinos funcionando las 24 horas del día.

Los contaminantes del aire como el dióxido de carbono (CO₂) y el monóxido de carbono (CO) pueden acumularse y superar los niveles permisibles, afectando negativamente la salud humana. La situación es aún peor en espacios cerrados, como las salas de fumadores de los casinos, donde no siempre hay una ventilación adecuada. Además, la falta de un sistema de monitoreo eficaz dificulta una intervención rápida cuando la calidad del aire se deteriora.

El problema principal del tema en mención es de la mala calidad del aire que afecta tanto a los consumidores como a los trabajadores, ya que tienen que trabajar continuamente en espacios interiores durante ocho horas a más y están expuestos pasivamente a esta contaminación.

1.2. Formulación del problema

De la descripción del problema surgen las siguientes interrogantes de investigación: Del problema principal

¿Cómo diseñar e implementar un prototipo IoT para la medición y monitoreo continuo de la calidad del aire en las salas de fumadores de un casino?

De los problemas específicos

- a. ¿Cuáles son los parámetros ambientales críticos que afectan la calidad del aire en las salas de fumadores de un casino?

- b. ¿Qué dispositivos y sensores IoT permiten una medición precisa y en tiempo real de la calidad del aire en ambientes cerrados?
- c. ¿Cómo se puede integrar un sistema IoT de monitoreo en tiempo real con mecanismos de control ambiental para mejorar las condiciones en las salas de fumadores?

1.3. Justificación e Importancia

El presente proyecto tiene como objetivo principal el monitoreo de la calidad del aire en las salas de un casino, implementando un prototipo de monitoreo para la detección de gases contaminantes en el aire dentro de un espacio cerrado, en este espacio y con ayuda de las tarjetas IOT y sensores modulares se detectara los niveles altos de Co2, limitando el tiempo que una persona pueda estar en este espacio y a su vez emitiendo una alerta en la que se activen los protocolos de seguridad como la extracción del aire de la zona contaminada y en el peor de los casos la salida del personal dentro de la sala de casino

El personal a cargo de este monitoreo podrá observar en tiempo real los niveles de Co2 permitidos en las salas del casino para iniciar una temprana acción a la eliminación de los contaminantes en el aire, también de manera remota se podrá ejecutar estos protocolos de seguridad gracias a las tarjetas IoT

Esta investigación no sólo tiene implicaciones para mejorar la salud de los fumadores y los trabajadores de casinos, sino que también podría ayudar a crear un modelo de vigilancia con innovación tecnológica que pueda aplicarse a otros entornos con riesgos similares.

Desde una perspectiva social este prototipo representa un avance importante en las zonas de entretenimiento al público, con ayuda de la implementación de tecnologías inteligentes ayudara a reducir las tasas de contaminación en la población, contribuyendo en reducir enfermedades respiratorias, reducir el impacto y la contaminación en el aire que recibe la capa de ozona día a día, mejorando también la concurrencia de personas a las salas de casino ya sea para personas sanas y/o con problemas respiratorios para su propio beneficio y entretenimiento.

Desde una perspectiva tecnológica, el desarrollo de un sistema de monitoreo de la calidad de aire en las salas de un casino es una oportunidad para mejorar las

tecnológicas existentes, este proyecto tiene como finalidad mostrar avances tecnológicos con ayuda de las tarjetas IoT y sensores que proporcionan soluciones avanzadas en la automatización con dispositivos IOT de manera remota y en tiempo real, así como para casinos y demás centros de entretenimiento en el sector turístico.

Desde una perspectiva económica, el monitoreo de la calidad de aire supone una inversión de bajo costo y una mejora en los ingresos económicos a las salas de casino, como consecuencia de la implementación del prototipo se benefician en la concurrencia de personas a salas de casino que garantizan la salud de sus clientes, aumentando ganancias en proyecciones a largo plazo.

1.4. Objetivos

1.4.1. Objetivo general

Diseñar e implementar un prototipo IoT para la medición y monitoreo de calidad de aire en las salas de fumadores de un casino.

1.4.2. Objetivos específicos

- a. Determinar los parámetros ambientales críticos que afectan la calidad del aire en las salas de fumadores de un casino
- b. Determinar los dispositivos y sensores IoT permiten una medición precisa y en tiempo real de la calidad del aire en ambientes cerrados
- c. Integrar un sistema IoT de monitoreo en tiempo real con mecanismos de control ambiental para mejorar las condiciones en las salas de fumadores.

1.5. Hipótesis

Por tratarse de una investigación aplicada, no se requiere en este estudio plantear hipótesis.

CAPÍTULO II: MARCO TEÓRICO

2.1. Antecedentes de la investigación

2.1.1. Antecedentes internacionales

Othman, H., Azari, R. y Guimarães, T. (2024) en su investigación “Monitoreo económico de la calidad del aire interior basado en IoT a bajo costo” manifiestan que su objetivo es diseñar, desarrollar y prototipar un sistema de monitorización de la Calidad del Aire Interior (CAI) pequeño y económico que pueda servir como una red de sensores basada en el Internet de las Cosas (IoT) para la evaluación de la ocupación de edificios. En este proyecto se construyó un dispositivo multisensor estacionario capaz de monitorizar y visualizar en tiempo real seis indicadores de CAI, con potencial para conectarse a dispositivos similares en el entorno construido.

Guatapi y García (2022) en su trabajo de investigación Diseño e implementación de un prototipo para un sistema de medición, análisis y purificador de gases contaminantes en el aire utilizando Arduino y UBIDOTS IoT, toman énfasis en la contaminación del aire en el exterior y el interior y para contrarrestar los efectos que causan estos gases a la salud en ambientes interiores confinados y cerrados, aplican su prototipo en una vivienda donde con la ayuda de sensores de gases puedan monitorear e identificar los niveles altos de gases contaminantes como el Co, amoníaco NH₃, ozono O₃, el nivel de ppm permitido a nivel 2.5 y 10 y a su vez extraer el aire contaminado a través de un ventilado extractor diseñado con un filtro capaz de retener las partículas contaminantes y solo expulsar el aire limpio.

En el estudio realizado por Klepeis et al. (2016) “Medición de la calidad del aire interior e involucramiento de las partes interesadas indígenas de California en el Win-River Resort and Casino: desarrollo colaborativo de políticas libres de humo”, midieron la calidad del aire en casinos y concluyeron que las concentraciones de PM_{2.5} pueden alcanzar niveles peligrosos en zonas donde se permite fumar, con implicaciones directas en la salud ocupacional.

York, N. y Lee, K. (2010) en la investigación “Una evaluación de referencia de la calidad del aire en los casinos tras la promulgación de la Ley de Aire Limpio en Interiores de Nevada”, encontraron que, donde persiste fumar o divorcios de la norma, los niveles de PM_{2.5} siguen siendo superiores a los valores saludables de la EPA; en contraste, casinos/áreas 100% libres de humo mostraron reducciones substanciales.

La figura 1 muestra al prototipo basado en un sistema IoT desarrollado para realizar pruebas con el monóxido de carbono y medir los niveles de concentración que se emiten, dando como resultado la activación del extractor de aire.

Figura 1

Medición del monóxido de carbono CO con un sensor MQ-7



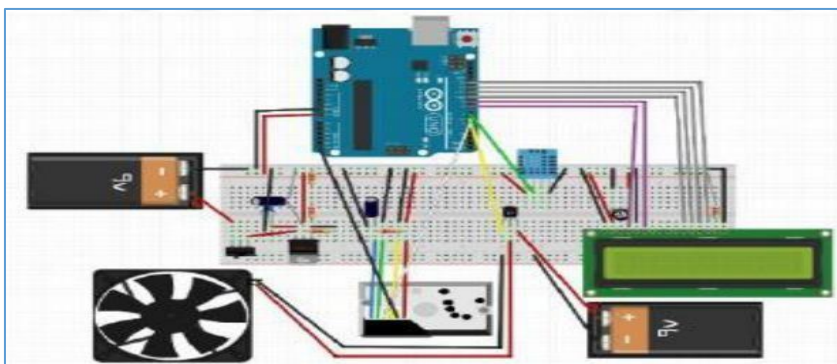
Nota. Adaptada de Guatapi y Garcia (2022) del sitio web <https://dSPACE.ups.edu.ec/bitstream/123456789/24088/1/UPS-GT004189.pdf>

2.1.2. Antecedentes nacionales

Zevallos (2024) en su trabajo de investigación “Modelo de monitoreo de la calidad de aire interior basado en arquitectura IOT” hacen énfasis en la medición del aire con el sensor MQ-135 en un laboratorio de una institución en Lima de educación superior donde recolectan datos por 3 meses de lunes a viernes y en intervalos diarios de 3-5 horas donde a través del módulo WIFI ESP8266 a la plataforma Thingspeak lo suben a la nube para realizar un pronóstico de la calidad del aire a futuro de tal forma que aseguran mejorar la salud y bienestar de las personas asociadas a la institución. Ver figura 2.

Figura 2

Diagrama del sistema propuesto



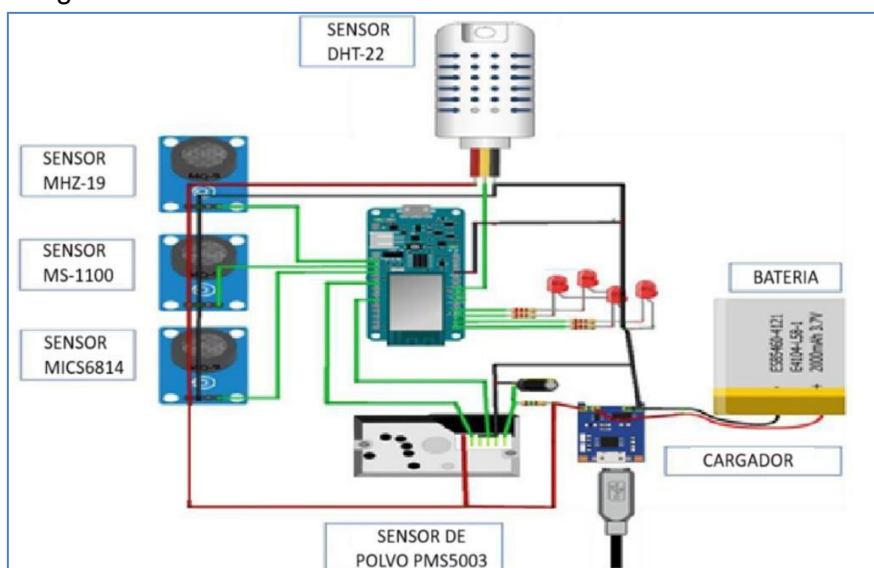
Nota. Adaptada de M. Zhou, A. Abdulghani, M. Imran y Q. Abbasi (2020) del sitio web <https://eprints.gla.ac.uk/214864/7/214864.pdf>

Mendoza y Cordero (2022) en su trabajo de tesis “Diseño de un sistema IoT para el monitoreo remoto de la calidad del aire en ambientes interiores de la Facultad de Ingeniería de la Universidad Privada de Tacna”, analizaron la calidad del aire utilizando una topología tipo malla de la topología tipo Malla. Demostraron la viabilidad técnica del sistema IoT (selección de sensores, topologías de red, plataformas cloud) y presentan datos de prueba en entornos académicos. Como resultado se tuvo un buen rendimiento en la red Malla con valores óptimos de RSSI (> -60 dBm). Los registros se mostraron en la interfaz gráfica dando a conocer que los ambientes probados están dentro de los estándares de calidad de ambiente (ECA) del aire máximos permitidos.

Medina de la Cruz (2020) en su trabajo de investigación “Diseño e implementación de un dispositivo de Internet de las cosas (IoT) para la medición y monitoreo de la calidad de aire en zonas urbanas” se enfocó en resolver el problema de la contaminación en el aire en zonas urbanas basándose en datos recopilados de la (OMS), indicando que el 92% de personas a nivel mundial inhalan aire contaminado por las elevadas concentraciones de gases del efecto invernadero. Su proyecto se basó en diseñar un sistema IoT para medir y monitorear el aire usando Arduino MKR1000 y el protocolo MQTT para subirlo a la nube y almacenar los reportes y diagramas de alerta diario en tiempo real en una base de datos. Ver figura 3.

Figura 3

Diagrama del sistema IOT basado en Arduino



Nota. Adaptada de M. Medina de la Cruz (2020) del sitio web https://repositorio.ucl.edu.pe/bitstream/handle/20.500.12872/506/Medina_MR_tesis_ingenieria_electronica_2020.pdf?sequence=1&isAllowed=y

2.2. Bases teóricas

2.2.1. Calidad del aire

La calidad del aire depende de la cantidad de contaminantes en el aire que respiramos y puede variar dependiendo de muchos factores como la actividad humana, la ventilación de las habitaciones y el uso de materiales contaminantes. Los contaminantes más comunes que se encuentran en las salas de fumadores de los casinos son:

- *Dióxido de carbono (CO₂)*: se produce principalmente por la quema de cigarrillos y la respiración humana. Altas concentraciones de CO₂ pueden causar fatiga, dolores de cabeza y en concentraciones muy altas, incluso pueden causar mareos o pérdida de conocimiento.
- *Monóxido de carbono (CO)*: Es un gas tóxico que también se produce al quemar cigarrillos. La enfermedad afecta al cuerpo reduciendo la capacidad de la sangre para transportar oxígeno, lo que puede causar síntomas como mareos, dificultad para respirar y confusión.
- *Material particulado (PM_{2,5} y PM₁₀)*: Son partículas diminutas suspendidas en el aire que pueden inhalarse profundamente en los pulmones, aumentando el riesgo de enfermedades respiratorias y cardiovasculares.

La Organización Mundial de la Salud (OMS) establece niveles seguros para estos contaminantes, recomendando concentraciones de CO₂ por debajo de 1.000 ppm, de CO por debajo de 9-ppm y de PM_{2,5} y PM₁₀ en línea con los estándares locales de calidad del aire. La Tabla 1 muestra los valores de los límites permitidos por la OMS para diferentes compuestos contaminantes del aire en tiempos establecidos.

Tabla 1

Límites permitidos por la OMS

Compuesto	Límites permitidos por la OMS			
	15min	1 hora	8 horas	24 horas
CO ₂	> 1000 ppm			
CO	100 mg/m ³	35 mg/m ³	10 mg/m ³	4 mg/m ³
PM _{2.5}	Media anual 5 µg/m ³			15 µg/m ³
HCHO	2 ppm	-	0.75 ppm	0.016 ppm

Nota. Adaptado de valores límite de gases, partículas y contaminantes en el aire que respiramos según la OMS del sitio web <https://es.air-q.com/grenzwerte>

2.2.2. Internet de las cosas (IoT)

En 1999, Kevin Ashton, de Procter & Gamble, presentó por primera vez el término "Internet de las Cosas". En pocas palabras, el Internet de las Cosas (IdC) es una red de objetos que incorporan sensores, software y otras tecnologías relacionadas, principalmente para conectar, intercambiar y transferir datos con otros dispositivos y sistemas a través de internet. Estos dispositivos pueden abarcar desde simples objetos domésticos hasta complejas herramientas y tecnologías empresariales e industriales.

Según Risteska y Trivodaliev (2017), los objetos inteligentes capaces de comunicarse y realizar cálculos entre sí están presentes en todas partes a nuestro alrededor. Por ejemplo, desde sensores sencillos, electrodomésticos y teléfonos inteligentes sofisticados hasta dispositivos industriales. Estas redes heterogéneas de estos objetos/dispositivos se engloban en un fenómeno conocido como el Internet de las Cosas (IoT).

Vermesan y Friess (2013) explican que el Internet de las Cosas (IoT) es una red que permite que objetos heterogéneos se conecten en cualquier momento y lugar a través de internet.

En pocas palabras, IOT es una red de dispositivos físicos conectados a internet que pueden recopilar, intercambiar y procesar datos en tiempo real. En el contexto del monitoreo de la calidad del aire, los dispositivos IoT se utilizan para medir parámetros como la concentración de gases y partículas en el aire, la temperatura, la humedad y la velocidad del aire. Un ejemplo típico de aplicación de IoT en la monitorización ambiental es el uso de sensores conectados para medir niveles de contaminación como CO₂, CO y partículas finas (PM). Estos sensores transmiten datos en tiempo real a plataformas de análisis lo que permite realizar ajustes en los sistemas de ventilación y otros parámetros ambientales para optimizar la calidad del aire.

Según Zhou et al. (2020) La integración de IoT no solo facilita la recopilación de datos, sino que también permite la toma de decisiones automatizada para mejorar las condiciones ambientales sin la necesidad de una intervención manual constante.

Según Radoglou Grammatikis, Sarigiannidis y Moscholios (2019), el Internet de las Cosas (IoT) está compuesto por numerosas redes en las que los dispositivos pueden comunicarse entre sí a través de Internet. Estos dispositivos se denominan comúnmente

«cosas» y se muestran en la figura 4 a continuación. Cada una de estas «cosas» tiene su propio conjunto de atributos.

Figura 4
Atributos de los Dispositivos IoT



Nota. Adaptada de Panagiotis I., Grammatikis, R., Sarigiannidis P. y Moscholios L. (2019) del sitio web.

<https://www.sciencedirect.com/science/article/pii/S2542660518301161>

2.2.2.1. Componentes específicos del internet de las cosas (IoT)

En el marco del IoT, los sensores y dispositivos juegan un papel importante al recopilar datos del entorno. Los sensores son capaces de recopilar información como temperatura, presión, movimiento, y en su defecto fundamentales para la conexión de lo físico a lo digital, ejemplo son aquellos sensores de placa integrada en frigoríficos inteligente, ellos captan datos como temperatura y los envían a través de la nube para su interacción con el usuario final y su respectivo análisis.

a) La conectividad

En el IoT nos referimos a la tecnología y metodología que ayudan a la comunicación en red entre ciertos dispositivos y un sistema IoT. Incluye también la transmisión y recepción de datos por medio de una red, una parte importante para el funcionamiento de los dispositivos IoT

Existen variedad de opciones de conectividad que depende de la aplicación, estas alternativas van desde redes de banda celular hasta redes de área amplia de baja potencia (LPWA), estas ofrecen una mejor cobertura y duración de tiempo en batería. La selección de la tecnología tiene un impacto en la productividad del dispositivo, los precios operativos y el latente de la aplicación.

b) El Procesamiento de Datos

Dentro de los sistemas IoT involucra convertir datos recopilados por los sensores en información relevante. Este proceso es importante para que los datos sean productivos, permitiendo que los dispositivos realicen labores automatizadas o suministren información útil para los usuarios.

Las plataformas IoT usan algoritmos avanzados de procesamiento de datos para manipular amplios volúmenes de información, asegurando un análisis eficaz y minucioso.

Dentro de los sistemas IoT implica convertir los datos crudos recopilados por los sensores en información útil. Este proceso es esencial para que los datos sean aprovechables, permitiendo que los dispositivos realicen tareas automatizadas o proporcionen información relevante a los usuarios. Las plataformas IoT modernas emplean algoritmos avanzados de procesamiento de datos para manejar grandes volúmenes de información, asegurando un análisis eficiente y preciso.

c) La interfaz de usuario

En los sistemas IoT este es el punto de relación entre el usuario y sistema. Está diseñada para plasmar la información procesada de manera precisa y accesible, facilitando la supervisión y el control de dispositivos IoT

Las interfaces de usuario pueden cambiar desde aplicaciones móviles sencillas para la gestión de dispositivos hasta pantallas de difícil visualización de datos. El diseño de la UI en un sistema IoT debe considerar factores como la usabilidad, la capacidad de gestión y el acceso para asegurar una interacción de usuario fluida y adecuada.

2.2.3. Topología de red IoT

Revisando información de diferentes profesionales en redes IoT se observa que existen 3 modelos de topología para IoT.

- Modelo estrella
- Peer-To-Peer
- Mesh o Malla

La topología estrella es un diseño de red donde los dispositivos IoT se conectan a una central. Este nodo central da gestión al tráfico existente en la red donde si un dispositivo falla, no afecta a los demás, es más usado en tecnologías con Wi-fi. La desventaja de esta topología es que si falla el nodo central puede detener toda la red.

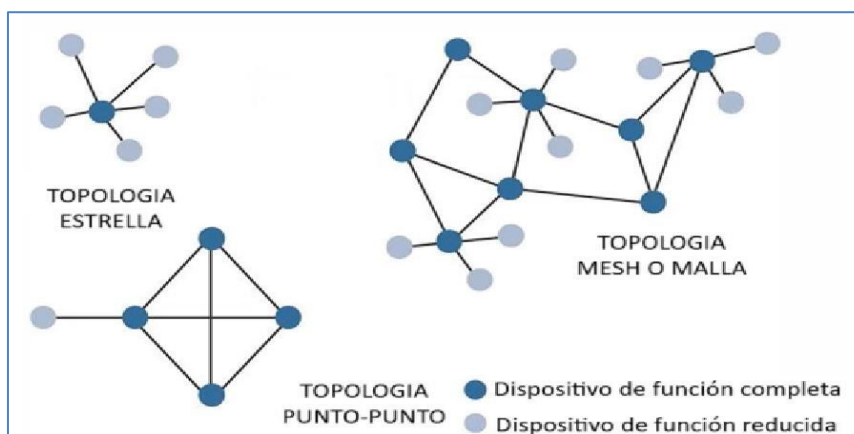
Para IoT la topología peer-to-peer es una arquitectura de red descentralizada donde cada dispositivo interactúa directamente entre sí, sin un nodo centralizado. Cada dispositivo actúa tanto como cliente como también servidor, comparten recursos como

ancho de banda e información. Mejora la fiabilidad y autonomía de la red IoT al depurar puntos de fallos único y poder tomar decisiones distribuidas.

Al igual que la topología peer to peer, la topología malla es descentralizada donde múltiples dispositivos interactúan entre sí de forma rápida y no jerárquica. Cada dispositivo IoT tiene la función de transmisión y retransmisión, lo que crea una red flexible y capaz de actuar a fallos dejando que la información viaje por vías diferentes hasta llegar a su punto de destino. Esta arquitectura es ideal para extender la cobertura inalámbrica, como protocolos Zigbee y Thread. La figura 5 muestra los tipos de topologías que se emplean para implementar redes IoT.

Figura 5

Tipos de topología de redes



Dentro de la topología de red IoT existente, hablaremos de un término que será nodo a los dispositivos dentro de la red que lo conforman. Existe diferentes tipos de nodos y depende enteramente de la función que cumpla:

- *Nodo final:* Es el dispositivo encargado de recopilar la información para su respectivo envío.
- *Nodo central:* Cumple una función similar a la de una interfaz de comunicaciones y es que es el dispositivo encargado de enviar y recibir la información de los nodos finales y transmitirla a una capa superior.
- *Nodo superior:* Son los dispositivos que cumplen la función de almacenamiento, procesamiento y plasmar la data de la cual ha sido programado.

2.2.4. Medios de transmisión

Se definen como las vías físicas por la que navegan los datos, se dividen en dos medios los cuales pueden ser alámbricos (cable de par trenzado, coaxial y fibra óptica) e inalámbricos (ondas de radio, microondas y luz infrarroja). Los medios alámbricos se caracterizan por ser una conexión más certera y puntual, en cambio los inalámbricos permiten la comunicación no guiada a través de señales aéreas.

En la figura 6 se observa el cable de par trenzado, que es el más usado comúnmente en hogares como en edificaciones de gran envergadura, es un tipo de cable que consisten en 4 pares de hilos de diferente coloración, con un revestimiento para reducir la interferencia electromagnética y la diafonía.

Figura 6
Tipos de medios guiados



Nota. Adaptada medios de Transmisión de UTEL (2020) del sitio web <https://apps.utel.edu.mx/recursos/files/r161r/w24365w/Medios%20de%20transmision%202.pdf>

La forma de su trenzado permite cancelar las ondas magnéticas, mejorando la calidad de la transmisión de datos. Existen 3 tipos de cables como lo son el UTP, el STP y el FTP, y se clasifican por categorías según el rendimiento y su ancho de banda.

- a. **Cable UTP:** Llamado así por las siglas Unshielded Twisted Pair (par trenzado no apantallado), este tipo de cable solo cuenta con una cubierta de PVC, con una impedancia de 100 Ohm. Tienen en común con el cable FTP el conector que es RJ45
- b. **Cable STP:** Es el cable en el que cada par de cobre está protegido por una malla metálica para evadir interferencias y ruido eléctrico, es más costoso que los otros cables, pero presenta muchas propiedades dependiendo del tipo de uso que se le dé como el industrial o minero, el conector usado en este tipo de cable es el RJ49.
- c. **Cable FTP:** Conocido por las siglas Foiled Twisted Pair (trenzado

con pantalla global), cuenta con par de hilos de cobre que no presentan apantallados, cubiertos con una pantalla de aluminio global para la protección física externa que pudiera recibir. Con una impedancia de 120 Ohm y una capacidad de transmisión similar a la del cable UTP.

Este tipo de cable con cuatro pares trenzados de hilo de cobre, tiene una conexión a un puerto llamado RJ45, provisto de 8 contactos planos para la conexión a un conector RJ45 usando las herramientas adecuadas para su ponchado.

Diseñado por la EIA (Electronic Industries Alliance), donde se definen 2 estándares que recopilan las normativas de instalación y diseño en cableado estructurado.

- *TIA/EIA 568-A*: Regula el uso para cableado estructurado en edificios, topologías, rendimiento, y los puertos que se deberían utilizar.
- *TIA/EIA 568-B y 568-B1*: Se definen los tipos de cables. Conectores a utilizar, características técnicas y de mantenimiento.

Para las dos normas, su sistema de conexión basados en la distribución de colores tiene diferente orden a lo que se denomina conexión directa y conexión cruzado como se detalla en las figuras 7 y 8.

Figura 7

Cable directo según TIA/EIA 568-A

Color	T568A	Telefonía	10/100BASE-T	1000BASE-T	PoE-A	PoE-B
Blanco/verde			TX+	Bidireccional	48V +	
Verde			TX-	Bidireccional	48V +	
Blanco/naranja			RX+	Bidireccional	48V -	
Azul		Ring		Bidireccional		48V +
Blanco/azul		Tip		Bidireccional		48V +
Naranja			RX-	Bidireccional	48V -	
Blanco/marrón				Bidireccional		48V -
Marrón				Bidireccional		48V -

Nota. Adaptada de Castillo (2020) en el artículo de Profesional Review: Cable par trenzado — Características, construcción, tipos y categorías, del sitio web <https://www.profesionalreview.com/2020/09/12/cable-par-trenzado-caracteristicas/>

Figura 8

Cable directo según TIA/EIA 568-B

Color	T568B	Telefonía	10/100BASE-T	1000BASE-T	PoE-A	PoE-B
Blanco/Naranja			TX+	Bidireccional	48V+	
Naranja			TX-	Bidireccional	48V+	
Blanco/Verde			RX+	Bidireccional	48V -	
Azul		Ring		Bidireccional		48V +
Blanco/azul		Tip		Bidireccional		48V +
Verde			RX-	Bidireccional	48V -	
Blanco/marrón				Bidireccional		48V -
Marrón				Bidireccional		48V -

Nota. Adaptada de Castillo (2020) en el artículo de Profesional Review: Cable par trenzado — Características, construcción, tipos y categorías, del sitio web <https://www.profesionalreview.com/2020/09/12/cable-par-trenzado-caracteristicas/>

Para la conexión directa se utiliza 2 nodos que son similares, ejemplo es una computadora con un switch. Estas conexiones se utilizan también en telefónica, en ocasiones con RJ45 o también con RJ11 para suministrar red y energía a cámaras IP y diferentes dispositivos.

Figura 9
















Distribución de colores para cable cruzado tipo 1

Color	T568A	T568B	100BASE-T
Blanco/verde			Blanco/Naranja
Verde			Naranja
Blanco/naranja			Blanco/Verde
Azul			Azul
Blanco/azul			Blanco/azul
Naranja			Verde
Blanco/marrón			Blanco/marrón
Marrón			Marrón

Nota. Adaptada de Castillo (2020) en el artículo de Profesional Review: Cable par trenzado - Características, construcción, tipos y categorías, del sitio web <https://www.profesionalreview.com/2020/09/12/cable-par-trenzado-caracteristicas/>

Figura 10

Distribución de colores para cable cruzado tipo 2

Color	T568A	T568B1	100BASE-T
Blanco/verde			Blanco/Naranja
Verde			Naranja
Blanco/naranja			Blanco/Verde
Azul			Blanco/marrón
Blanco/azul			Marrón
Naranja			Verde
Blanco/marrón			Azul
Marrón			Blanco/azul

Nota. Adaptada de Castillo (2020) en el artículo de Profesional Review: Cable par trenzado — Características, construcción, tipos y categorías, del sitio web <https://www.profesionalreview.com/2020/09/12/cable-par-trenzado-caracteristicas/>

El medio más usado es el par trenzado ya que se encuentra en todas las edificaciones posibles ya sea hogares o en la industria, minería y construcción. El protocolo usado para las redes LAN es el protocolo Ethernet, este protocolo permite comunicar diferentes dispositivos y maneja un estándar que sería la IEEE 802.3, con el paso de los años la tecnología para este medio ha ido evolucionando como se describe en la siguiente Tabla 2.

Tabla 2

Evolución del estándar 802.3xx respecto a la velocidad

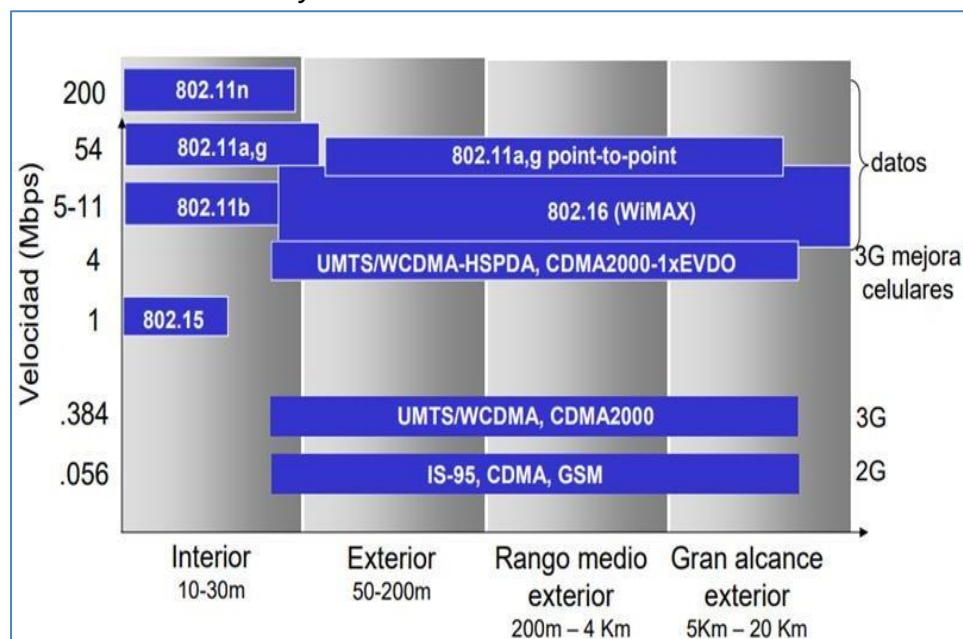
Año	IEEE	Nombre	Velocidad	Tipo de cable
1990	802.3i	10BASE-T	10 Mb/s	Cat-3
1995	802.3u	100BASE-T	100 Mb/s	Cat-5
1998	802.3z	1000BASE-SX	1 Gb/s	Fibra multimodo
		1000BASE-LX/EX	1 Gb/s	Fibra monomodo
1999	802.3ab	1000BASE-T	1 Gb/s	Cat-5e o superior
2003	802.3ae	10GBASE-SR	10 Gb/s	Laser-Opt. MMF
		10GBASE-LR/ER	10 Gb/s	Fibra monomodo
2006	802.3an	10GBASE-T	10 Gb/s	Cat-6A
2010	802.3ba	40GBASE-SR4/LR4	40 Gb/s	Laser MMF o SMF
		100GBASE-SR10/LR4/ER4	100 Gb/s	Laser MMF o SMF
2015	802.3bq	40GBASE-T	40 Gb/s	Cat-8 (Clase I y II)

El segundo tipo de medio de comunicación es el inalámbrico, por medio del cual no se usa un determinado medio físico para la transmisión, en su lugar la señal es enviada a través del aire por medio de determinadas frecuencias, algunas de las características del medio inalámbrico es la fuerza de la señal ya que algunas decrecen al cruzar materia (perdida de propagación). Las interferencias de otros orígenes como el compartir redes de una misma frecuencia con otros dispositivos, también puede influir el ruido magnético.

Una de las tecnologías en redes WLAN es el sistema wifi, esta tecnología se basa en estándares de la IEEE 802.11 y al igual que el medio alámbrico también ha ido progresando su capacidad para llevar información a través de la red desde distancias cortas hasta distancias mucho mayores como se muestra en la figura 11.

Figura 11

Demostración del rango de alcance que tienen los enlaces de radio a un determinado estándar y frecuencia



Nota. Adaptada de Kurose y Ross (2010) en Redes Inalámbricas y móviles, del sitio web <https://www.cartagena99.com/recursos/alumnos/apuntes/Tema%203-Redes%20inalambricas%20y%20moviles.pdf>

En la tabla 3 se muestra la evolución del Estándar 802.11xx con respecto a la velocidad de transmisión.

Tabla 3

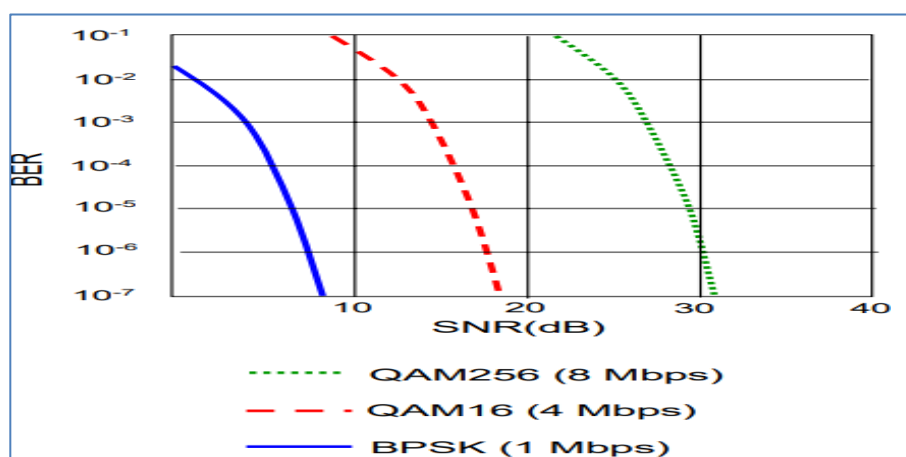
Estándar 802.11xx y su evolución respecto a la velocidad de transmisión

Año	IEEE	Generación	Velocidad	Frecuencia
1997	802.11	-	2 Mb/s	2.4 GHz
1999	802.11b	-	11 Mb/s	2.4 GHz
1999	802.11a	-	54 Mb/s	5 GHz
2003	802.11g	-	54 Mb/s	2.4 GHz
2008	802.11n	Wi-Fi 4	600 Mb/s	2.4 / 5 GHz
2014	802.11ac	Wi-Fi 5	6933 Mb/s	5 GHz
2019	802.11ax	Wi-Fi 6	9608 Mb/s	2.4 / 5 GHz
2020	802.11ax	Wi-Fi 6E	9608 Mb/s	2.4 / 5 / 6 GHz
2022	802.11be	Wi-Fi 7	40 Gb/s	2.4 / 5 / 6 GHz

Otra característica de las redes inalámbricas es la señal-ruido (SNR), donde el SNR mide la potencia de la señal frente al ruido, un mayor SNR facilita separar la señal del ruido, en donde tenemos una relación entre el SNR y el BER (tasa de errores de bit) donde a más potencia SNR menos BER y por ende menos ruido de esta forma mejoramos la calidad, confiabilidad y rapidez de la comunicación. En la figura 12 se demuestra la relación entre el SNR y el BER.

Figura 12

Relación señal SNR y BER



Nota. Adaptada de Kurose y Ross (2010) en Redes Inalámbricas y móviles, del sitio web <https://www.cartagena99.com/recursos/alumnos/apuntes/Tema%203-Redes%20inalambricas%20y%20moviles.pdf>

2.2.5. Recopilación de datos en redes IoT

La recopilación de datos es esencial para los dispositivos sensores, especialmente en tecnologías como el Internet de las Cosas (IoT), las redes de sensores inalámbricos

(WSN) y la nube de sensores (SC). En los últimos años, se ha publicado literatura en estas áreas de investigación para proponer diferentes modelos, arquitecturas y contribuciones.

Debido a la importancia de una recopilación eficiente de datos para reducir el consumo de energía, la latencia, la vida útil de la red y los costos generales, se ha publicado una importante cantidad de literatura para facilitar la recopilación de datos. Por lo tanto, se han realizado estudios de revisión sobre la recopilación de datos en estos dominios de forma aislada. Sin embargo, la falta de una revisión exhaustiva que identifique y analice colectivamente las diferencias y similitudes entre las propuestas de recopilación de datos en IoT, WSN y SC.

El objetivo principal de esta investigación es realizar un estudio exhaustivo para explorar el estado actual, los casos de uso, las contribuciones, las medidas de rendimiento, las medidas de evaluación y la arquitectura en los dominios de investigación de IoT, WSN y SC. Los hallazgos indican que los estudios sobre recopilación de datos en IoT, WSN y SC muestran resultados relativamente estables en los últimos cinco años.

2.2.6. Normativas sobre la calidad del aire

Las normativas y regulaciones sobre calidad del aire establecen los límites aceptables de concentración de contaminantes en el aire, con el fin de proteger la salud humana y el medio ambiente. Algunas de las normativas más relevantes son:

- *Normativa Europea de Calidad del Aire (2008/50/CE)*: Establece límites de calidad del aire para los principales contaminantes, incluidos CO₂, CO, PM_{2.5} y PM₁₀, y regula el monitoreo y la notificación de estos niveles en espacios cerrados y abiertos.
- *Directrices de la OMS (2018)*: La OMS ha proporcionado directrices que abogan por una mayor reducción de los niveles de contaminantes en espacios cerrados, especialmente en áreas como casinos, restaurantes y oficinas, donde la ventilación puede no ser adecuada.
- *Normativa Nacional de Calidad del Aire*: Dependiendo del país, existen leyes locales que regulan específicamente la calidad del aire en espacios cerrados. Estas regulaciones establecen umbrales de concentración para CO₂, CO y partículas, y en muchos casos exigen la implementación de tecnologías de monitoreo.

Estas normativas son fundamentales para guiar el diseño y la implementación de

sistemas de monitoreo y control de la calidad del aire en el casino, para asegurar que las condiciones del aire sean seguras tanto para los fumadores como para el personal.

2.2.6.1. Normativas sobre la calidad del aire en el Perú

En Perú, las regulaciones y leyes que rigen las operaciones de los casinos, particularmente aquellas relacionadas con la calidad del aire, el control del humo y las condiciones dentro de estos establecimientos, se centran principalmente en el control del tabaco y las normas ambientales.

A continuación, se presentan los aspectos más importantes a tener en cuenta en nuestra tesis sobre el monitoreo de la calidad del aire en las salas de fumadores de los casinos, norms que se encuentran resumidas en la Tabla 4.

a) Ley N° 29838 – Ley Antitabaco (2012)

Esta ley es fundamental para tu investigación, ya que regula el **consumo de tabaco** en espacios públicos, incluidos los casinos.

- *Prohibición de fumar en espacios cerrados:* La ley prohíbe fumar en la mayoría de los espacios cerrados públicos, pero permite la creación de áreas específicas para fumadores, las cuales deben estar **aisladas** para evitar la exposición al humo de otras personas.
- *Zonas habilitadas para fumadores:* En los casinos, se debe habilitar un área específica para fumadores, que esté bien ventilada y aislada de otras áreas donde se pueda generar un ambiente libre de humo para no fumadores.
- *Control de la exposición al humo de segunda mano:* La ley tiene como objetivo proteger a los no fumadores del humo de cigarro, que está asociado con varios problemas de salud, por lo que establece medidas para reducir la exposición.

b) Ley N° 28053 – Ley General para la Prevención y Control de la Contaminación del Aire

Esta ley establece las bases para el control de la contaminación del aire en Perú. Aunque no está enfocada exclusivamente en los casinos, sus disposiciones afectan a todos los espacios cerrados donde se genera contaminación del aire, incluyendo los salones de fumadores en los casinos.

- *Control de contaminantes atmosféricos:* La ley regula los niveles de

contaminantes en el aire, tales como partículas finas (PM2.5 y PM10), monóxido de carbono (CO) y otros gases que son emitidos por el humo del tabaco.

- *Monitoreo de la calidad del aire:* Establece que los responsables de establecimientos cerrados deben tomar medidas para garantizar que los niveles de contaminación no superen los límites establecidos por las autoridades de salud.

c) Reglamento de la Ley N° 29838 – Reglamento de Control del Consumo de Tabaco

Este reglamento regula el cumplimiento de la Ley Antitabaco, especificando cómo deben implementarse las zonas de fumadores en los establecimientos, incluidos los casinos.

- *Zonas para fumadores:* El reglamento detalla cómo deben ser las zonas para fumadores dentro de los casinos, en cuanto a ventilación, dimensiones y aislamiento.

- *Requisitos para la ventilación:* Las salas de fumadores deben cumplir con estándares específicos de ventilación para evitar que el humo se mezcle con el aire de otras áreas del casino, protegiendo así la salud de los no fumadores.

-

d) Norma Técnica Peruana NTP 399-014-2009 – "Calidad del aire – Directrices para la medición de la concentración de partículas en el aire"

Esta norma se enfoca en la medición de la calidad del aire, particularmente de las partículas en suspensión, que son uno de los principales componentes del humo del tabaco.

- *Medición de PM2.5 y PM10:* La norma establece los métodos y procedimientos para medir las concentraciones de partículas finas que se generan en ambientes cerrados, como las salas de fumadores de los casinos.

e) Reglamento del Ministerio del Ambiente (MINAM) sobre los estándares de calidad del aire

El Ministerio del Ambiente (2019) regula y supervisa la calidad del aire en el país. El reglamento del MINAM establece los límites máximos permisibles de

contaminantes en el aire, que incluye gases y partículas en suspensión generados por el humo de cigarro.

- *Monitoreo y control ambiental:* El MINAM también promueve el uso de tecnologías y sistemas de monitoreo para garantizar que los establecimientos cumplan con los estándares ambientales.

f) Norma Técnica N° 20 – Calidad del Aire

Establece los límites máximos permisibles para varios contaminantes del aire, como monóxido de carbono (CO), óxidos de nitrógeno (NOx), dióxido de azufre (SO₂), ozono (O₃) y, por supuesto, partículas finas (PM_{2.5} y PM₁₀), las cuales se encuentran en niveles elevados debido al humo del tabaco.

- *Normas para espacios cerrados:* Aunque la norma no está diseñada específicamente para los casinos, puedes utilizarla como base para establecer un marco de calidad del aire en los espacios cerrados donde se fuma.

Tabla 4

Normas y Leyes relacionadas con la calidad del aire en Perú

Norma / Ley	Descripción	Aplicación en Casinos
Ley N ° 29.838 - Ley Antitabaco	Regula el consumo de tabaco en lugares públicos y permite áreas de fumadores aisladas	Establece las zonas de fumadores en los casinos, con requisitos de ventilación y
Ley N° 28053 - Ley General para la Prevención y Control de la Contaminación	Establece las regulaciones para el control de la contaminación del aire, incluidos los límites de	Asegura que las salas de fumadores cumplan con los estándares de calidad del aire establecidos
NTP 399.014:2009	Norma técnica que regula la medición de partículas finas en el aire (PM _{2.5} y PM ₁₀)	Se utiliza para medir los contaminantes en las salas de fumadores de los casinos, asegurando que no se
Reglamento del MINAM	Establece los estándares de calidad del aire en el Perú	Aplicable para monitorear la calidad del aire en los espacios cerrados, como las <u>salas de fumadores</u>

La Tabla 5 muestra los contaminantes del aire en la sala de fumadores según las normas vigentes en el Perú.

Tabla 5

Contaminantes del aire en la sala de fumadores

Contaminante	Descripción	Fuente Principal	Impacto en la Salud
PM2.5	Partículas finas suspendidas en el aire, de menos de 2.5 micrómetros	Humo de cigarro, emisiones de vehículos	Afecta los pulmones, aumenta el riesgo de enfermedades respiratorias y cardiovasculares
Monóxido de carbono	Gas inodoro y tóxico producido por la combustión incompleta	Fumar cigarro	Reduce la capacidad de oxigenación de la sangre, causa dolores de cabeza y mareos
Compuestos Orgánicos Volátiles (COVs)	Gases que se liberan al fumar y que incluyen sustancias como el benceno y el formaldehído	Humo de cigarro	Irrita las vías respiratorias, puede causar cáncer a largo plazo
Óxidos de Nitrógeno (NOx)	Gases compuestos de nitrógeno y oxígeno, resultantes de la quema de tabaco	Humo de cigarro	Contribuye a la formación de ozono y partículas finas, perjudicando la salud respiratoria

2.3. Definición de términos

2.3.1. Internet de las cosas (IoT)

Internet de las cosas (IoT) es una red de objetos físicos equipados con sensores, software y otras tecnologías que pueden recopilar y transmitir datos a través de Internet sin intervención humana directa. Los dispositivos IoT permiten mediciones en tiempo real de la calidad del aire, por ejemplo: concentración de contaminantes (CO₂, CO,

PM2.5, PM10), temperatura y

humedad. Estos sensores transmiten datos a plataformas de análisis, proporcionando un monitoreo continuo y una toma de decisiones automatizada para optimizar las condiciones ambientales. El Internet de las cosas es esencial para monitorear la calidad del aire en espacios interiores como las salas de fumadores de los casinos, ya que permite el monitoreo continuo y el ajuste automático de la ventilación u otros sistemas de control ambiental (Chacón y Cardona, 2023).

2.3.1. Calidad del aire

La calidad del aire es una medida de la concentración de contaminantes en el aire y sus efectos sobre la salud de los organismos vivos. Este término engloba una serie de factores, incluida la presencia de gases nocivos (como dióxido de carbono, monóxido de carbono) y partículas finas (como PM2.5 y PM10), que pueden tener efectos negativos en la salud humana. En espacios cerrados, como las zonas de fumadores de los casinos, la calidad del aire puede verse afectada debido a la falta de ventilación adecuada, lo que provoca una acumulación de contaminantes. El monitoreo de la calidad del aire mediante sistemas IoT es importante para garantizar entornos saludables y seguros, minimizando los riesgos para los fumadores y los empleados que trabajan en estos lugares (Quishpe y Toapanta, 2023).

CAPÍTULO III: MARCO METODOLÓGICO

3.1. Diseño de la investigación

El tipo de investigación desarrollada en esta tesis es aplicada o práctica., donde la importancia radica en que a través del diseño de un prototipo IoT se mida y monitoree la calidad de aire en las salas de fumadores de un casino.

La investigación es de naturaleza experimental ya que está conectado directamente a un entorno del mundo real (una sala de fumadores de un casino), lo que permite observar el impacto de la implementación de un sistema de monitoreo de la calidad del aire utilizando tecnología IoT. Este diseño permitió evaluar los cambios en la calidad de aire resultantes de la implementación del sistema comparando las condiciones antes y después de la intervención.

3.2. Acciones y actividades

La Tabla 6 muestra el cronograma de actividades desarrollado en la presente tesis.

Tabla 6

Cronograma de actividades

Actividad	Mes1	Mes2	Mes 3	Mes4	Mes5
Revisión de literatura	X				
Diseño y planificación del sistema IoT	X	X			
Compra y configuración de sensores		X			
Desarrollo de la plataforma de visualización			X	X	
Instalación del sistema en el casino				X	
Recolección de datos y análisis				X	X
Redacción de la tesis					X

3.2.1. Criterios iniciales

Para el desarrollo del proyecto se realizó el estudio de la zonificación del casino encontrando zonas denominadas “sala de fumadores” asignadas especialmente a los usuarios consumidores de Tabaco, salas cerradas con poca ventilación, siendo entonces los espacios indicados para el monitoreo de la calidad de aire, por lo que no es necesario cubrir toda la infraestructura del casino.

Se decidió desarrollar el proyecto con la topología que más se adecua a las necesidades

es la de tipo malla, que ofrece una gran redundancia y fiabilidad. En este tipo de topología, cada equipo está conectado a todos los demás mediante enlaces separados, Esta configuración proporciona caminos redundantes en toda la red, de modo que, si un enlace falla, otro puede encargarse del tráfico.” (MCSE Training Kit: Networking Essentials Plus, Microsoft Press, 2001), teniendo claro que en otras topologías tenemos algunas desventajas de fiabilidad y costos para nuestra red. La figura 13 muestra la parte posterior del casino de acceso a la sala de fumadores.

Figura 13

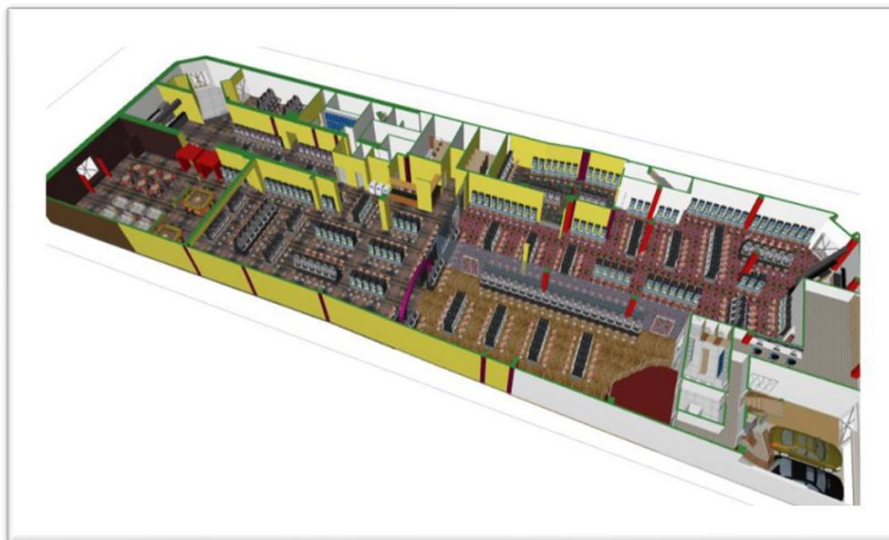
Casino parte posterior acceso a sala de fumadores



La figura 14 muestra los diferentes ambientes de la sala de fumadores del Casino.

Figura 14

Casino parte posterior de la sala de fumadores



La figura 15 muestra la distribución de las salas 5 y 6 de la sección de fumadores en el casino.

Figura 15

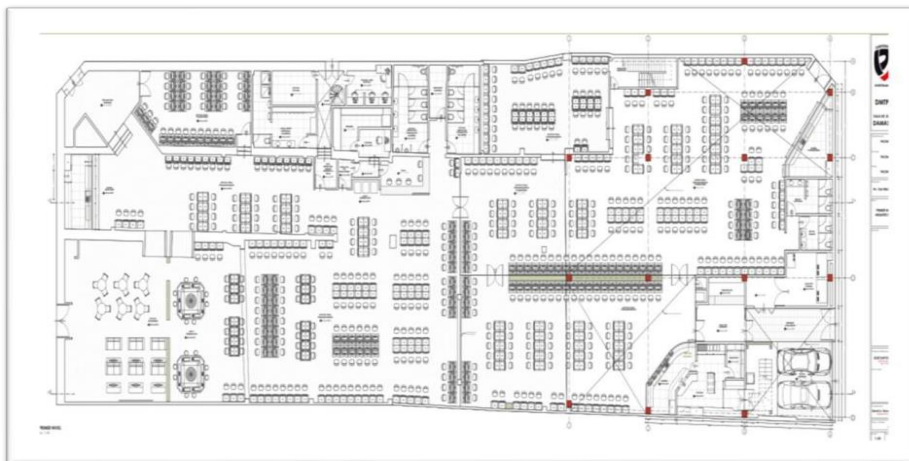
Salas de fumadores denominadas sala 5 y sala 6



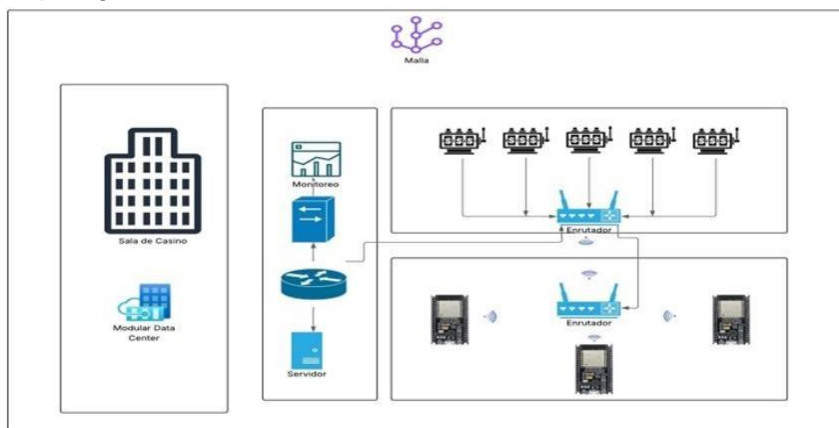
La figura 16 muestra la distribución de ambientes en el casino utilizando el software AUTOCAD.

Figura 16

Casino plano en AUTOCAD



La figura 17 muestra la topología de redes mesh. Está topología es la que más se adapta a nuestro diseño por su redundancia, fiabilidad y escalabilidad.

Figura 17*Topología de redes mesh*

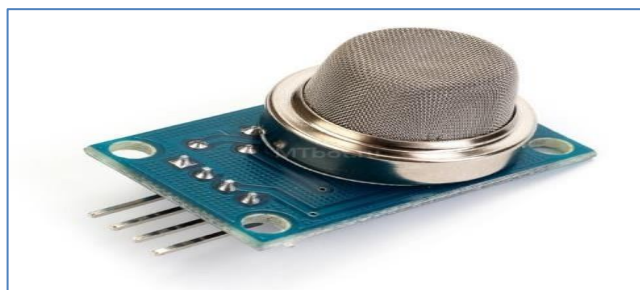
3.2.2. Selección de dispositivos

3.2.2.1. Sensores

La selección de los sensores de calidad de aire se realizó de acuerdo a la precisión que se requiere para medir los parámetros propuestos en los criterios señalados en nuestros indicadores tales como CO₂, Temperatura, Humedad. Partículas (PM_{2.5}).

Sensor de CO₂

Se optó por usar el MHZ19 sensor NDIR (infrarrojo no dispersivo) que mide la concentración de gas dióxido de carbono (CO₂), Sensor PMS5003 de Concentración de Partículas Alta Precisión PM_{2.5}, sensor DHT 22 de temperatura y humedad, SMAKN MS1100 Módulo de sensor de gas VOC Formaldehído Concentración de benceno Inducción de gas 100mA Breakout para Arduino, MiCS6814 Sensor de Gas CO.

Figura 18*Sensor de CO₂ modelo MHZ19*

Nota. Imagen disponible en https://ihormelnyk.com/mh-z19_co2_meter?srsltid=AfmBOopd_QG2wVGboVXCKp45LJsHi_cwR1WMuxqEFT3Y4kUW0Hz43q1Q

en https://ihormelnyk.com/mh-z19_co2_meter?srsltid=AfmBOopd_QG2wVGboVXCKp45LJsHi_cwR1WMuxqEFT3Y4kUW0Hz43q1Q

La tabla 6 muestra las características técnicas del sensor de CO₂ MH-Z19.

Tabla 7

Características técnicas de MH-Z19

Parámetro	Especificación
Modelo	MH-Z19 (variante A / B según versión)
Principio de medición	NDIR (Infrarrojo No Dispersivo)
Gas medido	Dióxido de Carbono (CO ₂)
Rango de medición	0 – 2000 ppm (algunas versiones hasta 5000 ppm)
Resolución	1 ppm
Precisión	±(50 ppm + 5% de la lectura)
Tiempo de respuesta	< 120 segundos (tiempo de respuesta completo)
Calibración	Automática y manual
Interfaz comunicación	UART (TTL), PWM, salida analógica (algunas versiones)
Voltaje de operación	4.5 – 5.5 V DC
Consumo de corriente	Promedio: 60 mA / Pico: 150 mA
Temperatura de trabajo	0 °C – 50 °C

Sensor de material particulado

“El PMS5003 es un sensor láser digital de alta precisión diseñado para medir la concentración de partículas suspendidas en el aire, específicamente PM1.0, PM2.5 y PM10, que son partículas de diferentes tamaños perjudiciales para la salud humana” (Unit Electronics, 2023). La figura 19 muestra al sensor PMS5003

Figura 19

Sensor de material particulado PMS5003



Nota. Imagen adaptada de Unit Electronics (2023), disponible en <https://uelectronics.com/producto/pms5003-sensor-de-calidad-de-aire-pm2-5/?srsltid=AfmBOorPNWV8eUav5xG7nq4s0RteTSIIJUyB3rbU31nLgipg-PLwO04J>

La Tabla 7 muestra las características técnicas del sensor particulado modelo PMS5003.

Tabla 8

Características técnicas de PMS5003

Parámetro	Especificación
Modelo	PMS5003
Fabricante	Plantower
Principio de medición	Dispersión de luz láser (conteo óptico de partículas)
Parámetros medidos	Concentración de partículas PM1.0, PM2.5 y PM10
Rango de medición	0 – 500 $\mu\text{g}/\text{m}^3$
Resolución	1 $\mu\text{g}/\text{m}^3$
Precisión	$\pm 10\%$ (dependiendo de condiciones ambientales)
Tiempo de respuesta	≤ 10 segundos
Voltaje de operación	4.5 – 5.5 V DC
Consumo de corriente	Promedio: 100 mA / Pico: 200 mA
Temperatura de trabajo	-10 °C – 60 °C
Humedad de trabajo	0 – 99 % RH (sin condensación)
Vida útil	≥ 3 años (dependiendo de condiciones de uso)

Sensor de temperatura y humedad

El sensor DHT22 es un sensor digital de temperatura y humedad relativa de bajo costo y excelente rendimiento. Según SAISAC Mecatrónica (2024) “está integrado por un sensor capacitivo de humedad y un termistor para medir el aire circundante, el cual muestra los datos mediante una señal digital en el pin de datos”. La Figura 20 muestra al sensor modelo DHT22.

Figura 20

Sensor de temperatura y humedad



Nota: disponible en <https://mecatronica.saisac.pe/producto/dht22-con-cable/>

La Tabla 8 muestra características técnicas del sensor DHT22 / AM2302.

Tabla 9

Características técnicas de DHT22 / AM2302

Parámetro	Especificación
Modelo	DHT22 / AM2302
Parámetros medidos	Temperatura y Humedad relativa
Rango de temperatura	-40 °C a +80 °C
Precisión de temperatura	±0.5 °C
Resolución de temperatura	0.1 °C
Rango de humedad	0 % – 100 % HR (sin condensación)
Precisión de humedad	±2 – 5 % HR
Resolución de humedad	0.1 % HR
Tiempo de respuesta	2 segundos (promedio)
Voltaje de operación	3.3 – 6 V DC
Consumo de corriente	1.5 mA en medición / < 0.5 mA en espera
Frecuencia de muestreo	0.5 Hz (1 medición cada 2 segundos)

Sensor de formaldehído modelo MS-1100

El sensor MS-1100 se emplea de modo general “para la detección de formaldehído, tolueno, el benceno y otros sensores de gas de COV. Tiene las ventajas de la alta sensibilidad y estabilidad, capaz de detectar más de 0.1ppm de gas, formaldehído, benceno, xileno y otros compuestos orgánicos volátiles” (CyD Tecnología, 2025).

Figura 21

Sensor de formaldehído modelo MS-1100



Nota. Imagen adaptada de <https://cdtecnologia.net/sensores/1328-sensor-de-gas-formaldehido-benceno-tolueno-cjmcu-1100-ms-1100-cov.html>

La Tabla 9 muestra las características técnicas de MS-1100 (CJMCU-1100).

Tabla 10

Características técnicas de MS-1100 (CJMCU-1100)

Parámetro	Especificación
Modelo / Nombre	MS-1100 (CJMCU-1100) — Sensor de Formaldehído y VOCs
Tipo de sensor	Sensor semiconductor de gases volátiles (VOCs)
Gases detectados	Formaldehído, benceno, tolueno, xileno, etanol, otros compuestos orgánicos volátiles
Rango de detección	Aproximado: 0 — 1000 ppm (detección cualitativa) (Amazon S3, Abra Electronics)
Voltaje de calentamiento	5 V ± 2%
Corriente de operación	Hasta 100 mA
Consumo de potencia	Máximo 430 mW (aprox.)
Señales de salida	Analógica (0–5 V) y digital (salida de comparación, LED de umbral) (Amazon S3, Arduino Learning, Keyestudio Wiki)
Tiempo de respuesta (T90)	< 5 s (tras precalentamiento de 3-5 minutos) (Amazon S3, Instructables)
Temperatura de operación	–10 °C a 60 °C (nominal ~20 °C) (Amazon S3, Abra Electronics)
Humedad de operación	Hasta 90 % RH (nominal ~65 % RH) (Amazon S3, Instructables)
Dimensiones	Aprox. 33 mm × 18 mm × 9 mm (±) (Keyestudio Wiki, Abra Electronics)
Peso	Aproximadamente 3.4 g (Keyestudio Wiki)
Ajuste de umbral	Potenciómetro onboard para calibrar el nivel de disparo digital (LED) (Arduino Learning, Keyestudio Wiki)

La Tabla 10 muestra las características técnicas del MICS-6814.

Tabla 11

Características técnicas del MICS-6814

Parámetro	Especificación
Modelo / Nombre	MICS-6814 (SGX Sensortech) — sensor de gas MOS multifunción
Gases detectables	CO: 1–1000 ppm
Principio de detección	Sensor de óxido metálico (MOS), con tres elementos independientes (RED, OX, NH ₃) (sgxsensortech.com, Sigma Robotics)
Dimensiones del chip	≈ 5 mm × 7 mm × 1.55 mm (sgxsensortech.com, Sigma Robotics)

Tabla 11 (continuación)

Parámetro	Especificación
Corriente de calefacción	RED: 32 mA / OX: 26 mA / NH ₃ : 30 mA (Siqma Robotics)
Resistencia en aire (R ₀)	RED: 100–1500 kΩ / OX: 0.8–20 kΩ / NH ₃ : 10–1500 kΩ (Siqma Robotics)
Rango operación (temp)	–30 °C a +85 °C (DigiKey, Siqma Robotics)
Humedad de operación	5 % – 95 % RH (sgxsensortech.com, Siqma Robotics)
Alimentación típica	4.9 — 5.1 V DC (sgxsensortech.com, DigiKey, Siqma Robotics)
Tipo de salida	Salida analógica (resistencia variable) — se requiere circuito de medida (ADC) (sgxsensortech.com, DigiKey)

3.2.2.2. Micro-controlador

EL ESP32 es un microcontrolador de bajo costo y bajo consumo energético, desarrollado por Espressif Systems, que integra en un solo sistema en chip (SoC) un procesador de alto rendimiento junto con conectividad inalámbrica WI-FI y bluetooth. Este dispositivo resulta especialmente adecuado para el desarrollo de soluciones en el ámbito del Internet de las Cosas (IoT), ya que ofrece una plataforma compacta, versátil y eficiente para la recolección, transmisión y procesamiento de datos. Ver figura 22.

El ESP32 se selecciona como unidad del control debido a sus características técnicas:

Procesamiento: incluye un microprocesador de doble núcleo Xtensa LX6, lo que permite gestionar simultáneamente múltiples sensores de gases y ambientales con tiempos de respuesta rápidos.

Conectividad inalámbrica: incorpora WI-FI (802.11 b/g/n) y bluetooth clásico/BLE, facilitando la comunicación directa con plataformas de visualización y control sin necesidad de hardware adicional, aspecto clave en aplicaciones IoT.

Bajo consumo energético: dispone de diversos modos de ahorro de energía, lo que lo hace apto para sistemas que pueden requerir funcionamiento autónomo o alimentado por batería.

Versatilidad en interfaces: cuenta con múltiples periféricos y pines de entrada/ salida (GPIO), incluyendo UART, I2C, SPI, PWM, ADC, lo cual permite la integración de distintos tipos de sensores ambientales como:

- MH-Z19 (CO₂, vía UART/PWM)

- PMS5003 (material particulado PM2.5, vía UART)
- DHT22 (temperatura y humedad, vía protocolo digital propietario)
- MS-1100 (formaldehído, vía señal analógica ADC)
- MICS-6814 (CO, NH₃, NO₂ vía ADC O I2C según el módulo)

Figura 22

Microcontrolador ESP32



Nota: Imagen adaptada de <https://microdesys.es/docs/microcontrolador-esp32/>

La posibilidad de integrar de manera directa y simultánea sensores que utilizan diferentes protocolos de comunicación representa una ventaja fundamental del ESPE32, ya que reduce la complejidad del diseño de hardware y permite centralizar el procesamiento de datos en un único dispositivo.

En consecuencia, el uso del ESP32 se ajusta plenamente a las necesidades del prototipo de monitoreo de calidad de aire, al proporcionar:

Una plataforma robusta y flexible para la adquisición de datos de múltiples contaminantes.

Conectividad inalámbrica integrada, que facilita la transmisión de información en tiempo real hacia aplicaciones de visualización y control.

Optimizaciones del consumo energético, lo que amplía sus posibilidades de aplicación en entornos con recursos limitados.

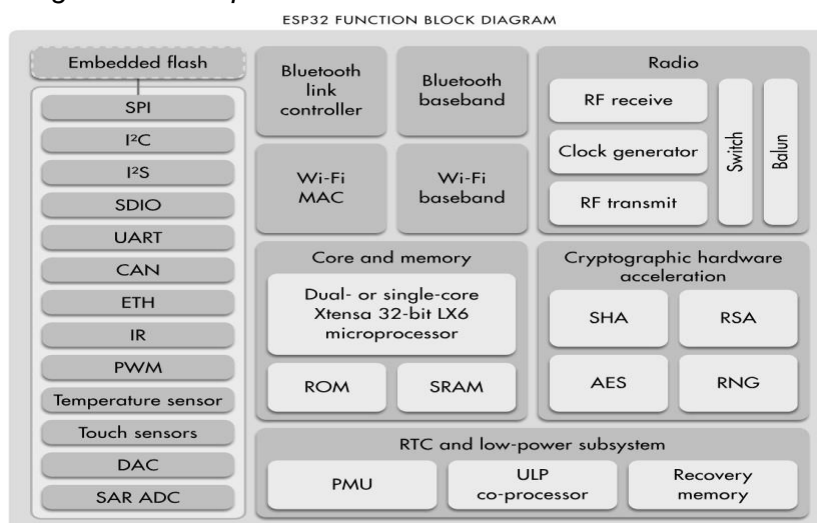
Por estos motivos el ESP32 contribuye a la base tecnológica idónea para la implementación de un sistema de monitoreo confiable, escalable y adaptable a diferentes escenarios de aplicación. La Tabla 11 muestra las características técnicas del ESP32.

Tabla 12*Características técnicas del ESP32*

Parámetro	Especificación
Modelo principal	ESP32 (serie de microcontroladores)
Arquitectura	Xtensa® Dual-Core 32-bit LX6 (a 160 o 240 MHz)
Memoria RAM	520 KB SRAM interna
Memoria Flash	Hasta 16 MB (dependiendo del módulo)
Conectividad inalámbrica	Wi-Fi 802.11 b/g/n (2.4 GHz), Bluetooth v4.2 BR/EDR y BLE
GPIO	34 pines de entrada/salida multipropósito
Interfaces soportadas	SPI, I ² C, I ² S, UART, CAN, ADC (12 bits, hasta 18 canales), DAC (2 canales de 8 bits)
Voltaje de operación	2.2 V – 3.6 V (típico: 3.3 V)
Consumo de energía	- Activo con Wi-Fi: ~160 mA - Deep-sleep: ~10 µA
Seguridad	Cifrado AES, SHA, RSA, ECC, RNG
Temperatura de operación	-40 °C a +125 °C
Dimensiones típicas módulo	25.5 mm x 18 mm (ESP32-WROOM-32)

La Figura 23 muestra el diagrama de bloques funcional del microcontrolador ESP32.

La etapa de conectividad WI-FI del ESP32 constituye una de sus principales características, ya que el microcontrolador integra de manera nativa un módulo inalámbrico compatible con el estándar IEEE 802.11 b/g/n. Esto le permite establecer enlaces de comunicación de alta velocidad dentro de redes locales (LAN) o directamente con dispositivos finales, sin requerir módulos externos adicionales.

Figura 23*Diagrama de bloques del módulo ESP32*

Nota. Figura referencial del ESP32

El microcontrolador ESP32 puede operar en diferentes modos de funcionamiento WI-FI:

Modo estación (STA, Station Mode): El dispositivo se conecta a un punto de acceso (router o AP), obteniendo acceso a la red local y, si está disponible, a Internet. Este modo es el más utilizado en aplicaciones de LoT, ya que facilita la transmisión de datos hacia servidores, plataformas en la nube o aplicaciones móviles.

Modo punto de acceso (AP, Access Point Mode): El ESP32 actúa como un punto de acceso WI-FI independiente, permitiendo que otros dispositivos (computadoras, smartphones, etc) se conecten directamente a él para intercambiar información sin necesidad de un router intermedio.

Modo mixto (AP + STA): Combina las dos modalidades anteriores, funcionando simultáneamente como cliente y como punto de acceso, lo que amplía su versatilidad en escenarios de monitoreo y control.

La integración de WI-FI en el ESP32 facilita la implementación de sistemas de monitoreo en tiempo real, ya que permite transmitir datos de sensores hacia plataformas de supervisión, aplicaciones de escritorio, aplicaciones móviles o servicios en la nube.

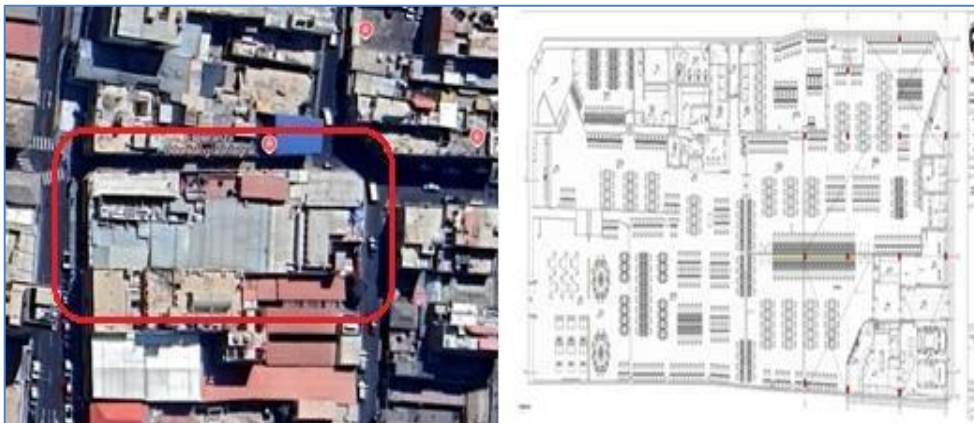
Asimismo, su compatibilidad con diferentes topologías de red (estrella, punto a punto y malla), lo convierte en un solución flexible y escalable para proyectos de internet de las Cosas (IoT).

En el presente prototipo de monitoreo de calidad de aire, la conectividad WI-FI del ESP32 resulta fundamental, puesto que posibilita:

- La comunicación inalámbrica entre el sistema de sensores y la aplicación de visualización.
- La integración directa con dispositivos móviles o computadoras, sin necesidad de cableado.
- La integración con redes existentes en el entorno de prueba (casino), optimizando la recopilación y el análisis de datos ambientales.

Figura 24

Área de cobertura para la aplicación del dispositivo IoT

**Figura 25**

Cobertura del módulo en el casino

**Figura 26**

Ubicación del AP y del prototipo IoT

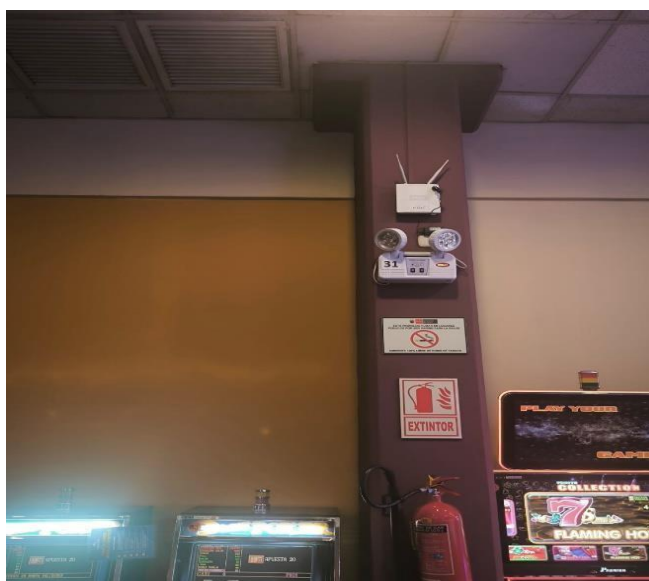
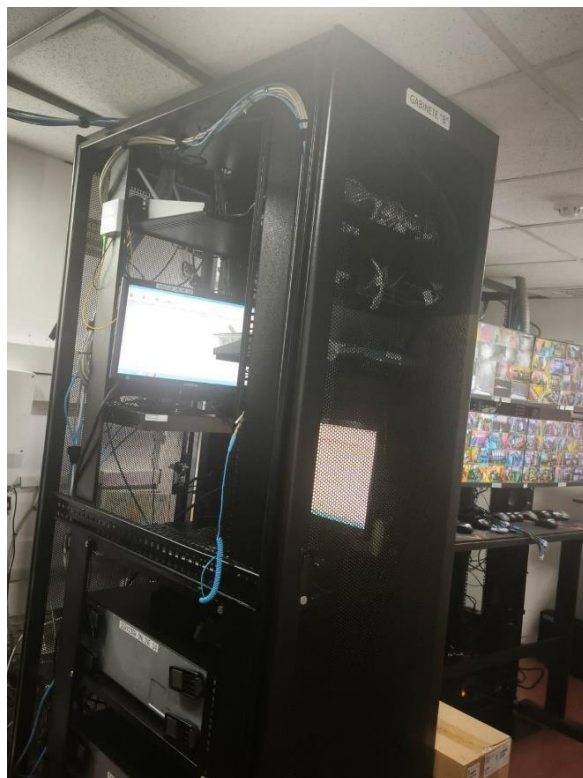


Figura 27

Sala de control para el monitoreo de la calidad de aire



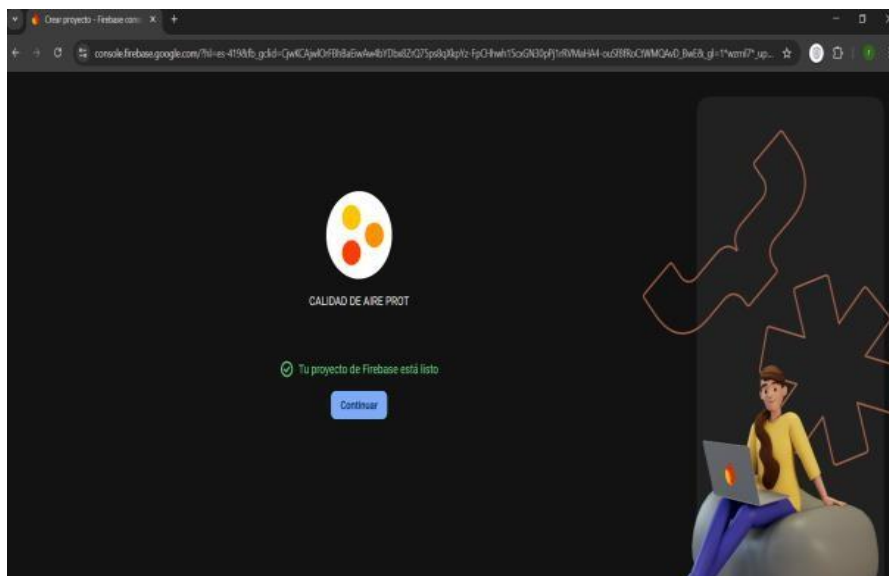
3.2.2.3. Interfaz gráfica

Firestore es una plataforma de desarrollo de aplicaciones en la nube creada por Google, que proporciona un conjunto de servicios integrados orientados al almacenamiento, procesamiento y sincronización de datos en tiempo real. Entre sus principales componentes destacan la base de datos en tiempo real (Realtime Database), Firestore, la autenticación de usuarios, el hosting de aplicaciones web y las notificaciones push, lo que permite construir soluciones escalables sin necesidad de gestionar servidores propios.

En particular, la Realtime Database y Cloud Firestore permiten que los datos sean almacenados y actualizados en la nube de forma inmediata, garantizando que todos los dispositivos conectados a la aplicación reciban la información de manera sincronizada y consistente. La Figura 28 muestra la utilidad de firestore en el monitoreo de calidad de aire en el casino.

Figura 28

Utilidad de firebase en el monitoreo de calidad de aire en el casino



En el presente prototipo de monitoreo de la calidad del aire en ambientes cerrados como un casino, Firebase se convierte en una herramienta clave al ofrecer:

Almacenamiento en la nube de los datos de los sensores: Las mediciones de CO₂, material particulado (PM2.5), temperatura, humedad, formaldehído y monóxido de carbono pueden ser enviadas desde el ESP32 directamente a Firebase mediante la conectividad Wi-Fi integrada.

Actualización en tiempo real: Los valores registrados por los sensores se reflejan instantáneamente en las aplicaciones vinculadas (por ejemplo, un panel de control en PC o en dispositivos móviles), lo que permite tomar decisiones rápidas en caso de detectar condiciones de riesgo ambiental.

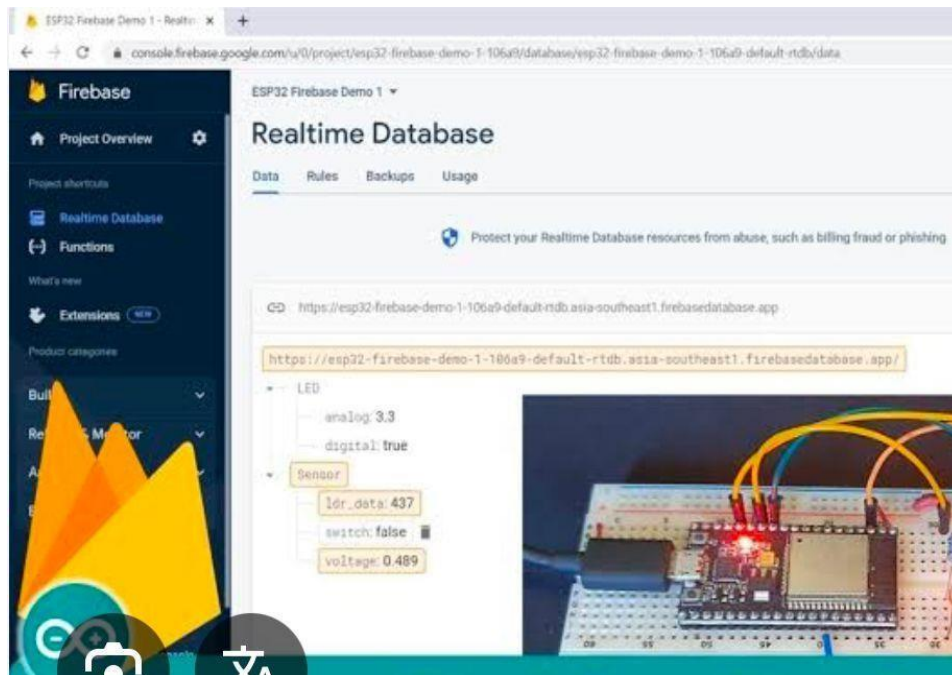
Acceso multiplataforma: La información almacenada en Firebase puede ser consultada desde diferentes dispositivos (computadoras, smartphones, tablets), lo que facilita la supervisión remota por parte del personal del casino o los encargados de mantenimiento.

Historial y análisis de datos: Firebase permite conservar registros históricos de las mediciones, lo que posibilita realizar análisis comparativos, generar reportes de calidad del aire y evaluar la efectividad de los sistemas de ventilación o purificación instalados en el ambiente.

Escalabilidad y flexibilidad: Al estar basado en la nube, Firebase permite ampliar el sistema fácilmente, integrando nuevos sensores o habilitando el monitoreo en múltiples zonas del casino sin necesidad de modificar la infraestructura local. La Figura 29 muestra el acceso a la base de datos en tiempo real de la plataforma web Firebase.

Figura 29

Plataforma web Firebase



3.2.2.4. Medidor de calidad de aire

El medidor de calidad de aire es un instrumento de medición que se usa para la toma de lecturas de los 6 parámetros principales para medir la calidad de aire en las salas de fumadores de un casino: Su propósito en el presente proyecto es el de tener como un punto de referencia la lectura correcta de los sensores del prototipo IoT, a su vez obtener una calibración precisa de los sensores del prototipo IoT.

El medidor de aire del fabricante Langder Technology Co., Ltd, es un detector multifuncional de calidad del aire que detecta formaldehído (HCHO), compuestos orgánicos volátiles totales (TVOC), partículas de menos de 2,5 micras (PM2.5/10), ICA, temperatura y humedad, con función de reloj y registro. Como dispositivo científico de detección de la calidad del aire, combina múltiples sensores de aire con un ventilador integrado para permitir la monitorización en tiempo real del formaldehído (HCHO), compuestos orgánicos volátiles totales (TVOC), PM2.5/10, ICA, temperatura y humedad

en su pantalla LCD digital, ver Tabla x que contiene las Especificaciones del medidor de calidad de aire

Tabla 13

Especificaciones técnicas del medidor de calidad de aire

Especificaciones del medidor de calidad de aire	Valor
Product Size	190 x 80 x 52mm
Product Weight	350g (12.35 oz)
Display Method	LCD Screen
Measure	CO ₂ , PM _{2.5} , PM _{1.0} , PM ₁₀ HCHO (Formaldehyde) TVOC. Temperature, Humidity, AQI
Detection method for CO ₂	Infrared (NDIR)
Detection method for PM	Laser Scattering
HCHO technology	Electrochemistry sensor
TVOC technology	Semiconductor sensor
Concentration Unit CO ₂	PPM
CO ₂ Measuring Range	400-5000PPM
Concentration unit for PM	ug/m ³
Concentration unit for HCHO and TVOC	mg/m ³
PM measuring range	0-999 ug/m ³
HCHO measuring range	0.001-1.999 Mg/m ³
TVOC measuring range	0.001-9.999 Mg/m ³
AQI range	0-500
Atmospheric Pressure	12.5 PSI-15.4 PSI
Sampling Time	1.5 Seconds
Temperature Range	-10°C-50°C (14°F-122°F)
Storage Temperature	-10°C-50°C (14°F-140°F)
Relative Humidity	20%-85%
Humidity Range	20%-85%
Charging	1000 mAh 5V DC
Battery	2000 mAh Lithium Ion
Power source	Rechargeable Lithium battery 5V DC power charging via micro USB port.

Nota. Adaptada de productos Instruments & Meters de la empresa Langder Technology Company Limited, disponible en: Made-in-China.com

La calibración del prototipo IoT se da en la programación del ESP32 a través de la plataforma Arduino que emplea el lenguaje de programación C++. En el programa se tienen las fórmulas para el CO y formaldehído, con la finalidad de que cada sensor pueda dar una lectura digital, esta lectura digital es igualado a un valor dentro del programa para que pueda dar las lecturas en la plataforma digital firebase. La figura 30 muestra al equipo medidor de aire.

Figura 30

Medidor de calidad de aire



Nota. Adaptada de productos Instruments & Meters de la empresa Langder Technology Company Limited, disponible en: Made-in-China.com

3.2.3. Desarrollo del prototipo

1. Desarrollo del hardware

La alimentación del sistema se realizó mediante una fuente modular con salida regulada de 5 V y 3.3 V, la cual proporciona la energía necesaria tanto para el ESP32 como para los módulos sensores, garantizando estabilidad y reduciendo el ruido eléctrico en las líneas de señal.

En la Tabla 12 se muestra el consumo máximo del circuito detallando los sensores y chip interno.

Cuando conectas varios dispositivos en paralelo (o en “parrilla”), no se suman los voltajes: todos reciben el mismo voltaje de su línea (rail), ya sea 5 V o 3.3 V.

Lo que sí se suma es la corriente (mA) que consume cada dispositivo de ese mismo rail.

Por eso se hace un cálculo por cada rail, sumando las corrientes de los equipos que usan ese voltaje.

Tabla 14

Consumo de corriente total del circuito

Sensor	Tensión de alimentación (V)	Corriente típica / trabajo (mA)	Notas / fuente
MH-Z19B (CO₂)	4.5 – 5.5 V (usar 5 V)	60 mA (promedio) — pico 150 mA	Datasheet MH-Z19B (Winsen / fabricante). (Winsen Sensor)
PMS5003 (PM2.5)	5 V (ventilador)	≤100 mA (activo)	Manual Plantower / Adafruit. (fan + sensor). (AQMD, Adafruit)
DHT22 / AM2302 (Temp/Hum)	3.3 – 5.5 V (recom.: 3.3 V con ESP32)	≈2.5 mA (máx. lectura)	Datasheet AM2302 / DHT22. (Mouser Electronics, Adafruit)
MS-1100 (HCHO / VOC)	5 V típ.	≈80–100 mA (power ≈380–450 mW)	Especificación módulo MS-1100 (módulos muestran 100 mA; datasheet indica ≈380–450 mW). (CDCK File Uploads, Instructables)
MICS-6814 (CO, NO₂, NH₃)	módulo: 3.3–5 V (según breakout)	≈32 mA (sensor núcleo)	Datasheet MiCS-6814 / ficha Mouser (corriente típica 32 mA). (SGX Sensor Tech, Mouser Electronics)
ESP32 (módulo Dev / WROOM)	3.3 V	≈190 mA (activo medio Wi-Fi) — pico ≈240–380 mA (TX)	Espressif datasheet (tabla modos RF: TX/RX consumos). Usamos valor típico 190 mA y pico hasta ~380–400 mA por seguridad. (Espressif)

Distribución de dispositivos por rail

- Rail de 5 V: MH-Z19B, PMS5003, MS-1100, MICS-6814 (todos estos trabajan a 5 V).
- Rail de 3.3 V: ESP32 y DHT22 (conviene poner el DHT22 en 3.3 V para evitar problemas de compatibilidad con las señales lógicas).

Consumo típico de cada dispositivo

- MH-Z19B → 60 mA

- PMS5003 → 100 mA
- MS-1100 → 100 mA
- MICS-6814 → 32 mA
- ESP32 → 190 mA (en promedio cuando está activo)
- DHT22 → 2.5 mA

Suma de corrientes por rail

En 5 V:

- 60 mA (MH-Z19B)
- +100 mA (PMS5003) = 160 mA
- +100 mA (MS-1100) = 260 mA
- +32 mA (MICS-6814) = 292 mA total En 3.3 V:
- 190 mA (ESP32)
- +2.5 mA (DHT22) = 192.5 mA total

Nota sobre el regulador de 3.3 V

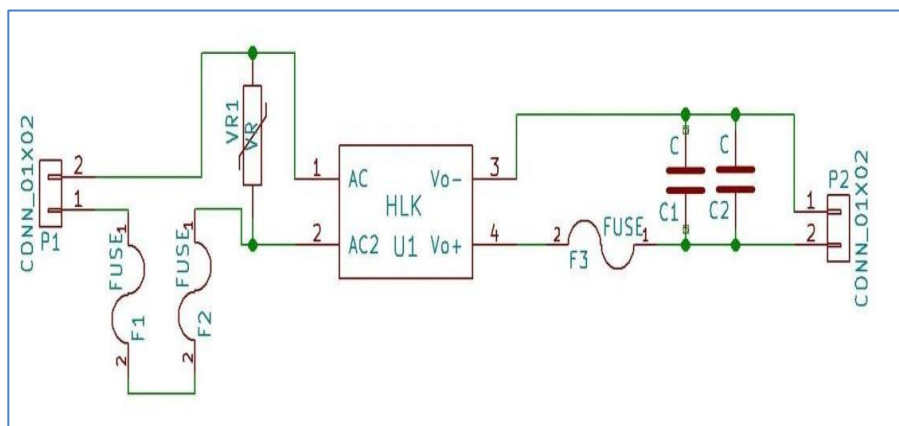
- Si la placa obtiene los 3.3 V a partir del rail de 5 V mediante un regulador:
- Con un LDO (lineal), la corriente que se extrae del rail de 5 V será casi igual a la corriente de salida (192.5 mA).
- Con un buck (convertor DC-DC), el consumo desde 5 V será menor porque aprovecha la eficiencia de conversión.

Para la alimentación del sistema se empleó un módulo AC-DC aislado Hi-Link HLK-PM05 que convierte 220 VAC a 5 VDC. Este módulo entrega hasta 5 W (≈ 1 A), admite 100–240 VAC, y tiene eficiencia alrededor de 70 % (típico), con encapsulado compacto y aislamiento de red (clase II). Desde este rail de 5 V se alimentan los sensores a 5 V y el cargador de batería.

La Figura 30 muestra el diagrama esquemático del circuito de la fuente de alimentación modular.

Figura 31

Diagrama esquemático del circuito de la fuente de alimentación



La Figura 31 muestra la imagen del circuito de la fuente de alimentación modular Hi-Link HLK-PM05

Figura 32

Fuente modular Hi-Link HLK-PM05



Para gestionar la batería Li-ion de 1 celda (3.7 V nominal, 4.2 V máx) y permitir carga y alimentación del sistema a la vez (power-path / load sharing), se utilizó un cargador MCP73871. Este controlador selecciona automáticamente la fuente entre entrada (5 V) y batería, permite fijar la corriente de carga hasta ~1 A, e incluye señales de estado (PG, STAT1/STAT2).

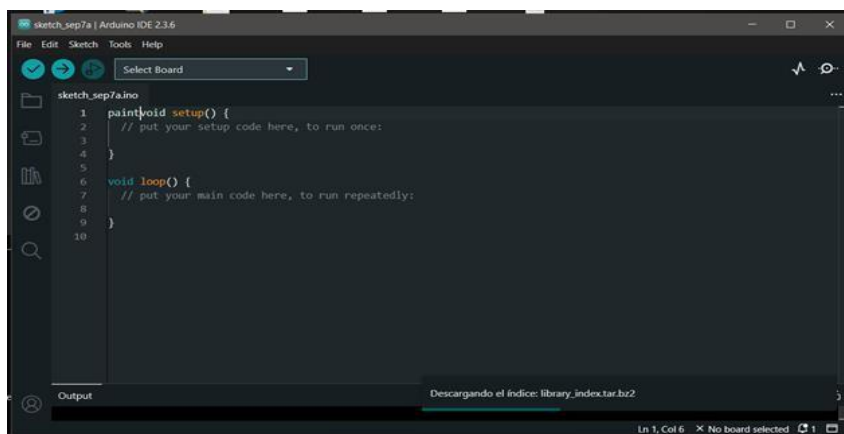
El rail de 3.3 V para el ESP32 se obtiene desde la batería (o desde 5 V cuando hay red) mediante un convertidor step-up/step-down 3.3 V (Pololu S7V8F3). Este módulo acepta 2.7–11.8 V de entrada, entrega 3.3 V regulados con eficiencia >90 % y

3.2.4. Desarrollo del software

En la etapa de desarrollo de software, el primer paso consistió en la programación de los microcontroladores. Para esta tarea es posible emplear diversos editores de texto, sin embargo, se optó por utilizar el entorno de desarrollo integrado (IDE) de Arduino, ya que permite tanto la transferencia del código fuente (sketch) al microcontrolador como la lectura del puerto serial a través de su terminal incorporada. Este IDE emplea el lenguaje de programación C++, ampliamente utilizado en el control de microcontroladores. Ver Figura 35.

Figura 36

Vista del entorno de IDE de Arduino



Inicializa comunicaciones (Serial, sensores), conecta el ESP32 a la red Wi-Fi y a Firebase, lee los sensores (DHT22, MH-Z19, PMS5003, MS-1100, MICS6814) y sube periódicamente esos valores a la base de datos de Firebase. Está pensado para ejecutarse en un ESP32 alimentado por la fuente Hi-Link.

Figura 37

Configuración inicial

```
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <DHT.h>
#include <SoftwareSerial.h>

// ----- CONFIGURACIÓN WiFi -----
#define WIFI_SSID "TuSSID"
#define WIFI_PASSWORD "TuPassword"

// ----- CONFIGURACIÓN FIREBASE -----
#define FIREBASE_HOST "https://TUPROYECTO.firebaseio.com/"
#define FIREBASE_AUTH "TOKEN_DE_FIREBASE"
FirebaseData fbd;
```

Librerías y definiciones (parte superior)

- #include <WiFi.h>: controla la conexión Wi-Fi del ESP32.
- #include <FirebaseESP32.h>: cliente para enviar/leer datos a Firebase Realtime Database.
- #include <DHT.h>: driver para el sensor DHT22 (temperatura/humedad).
- #include <SoftwareSerial.h>: crea puertos serie por software (usado para MH-Z19 y PMS5003).
- #define y FirebaseData fbdo: se declaran credenciales Wi-Fi y Firebase (HOST, AUTH) y el objeto fbdo que gestiona la comunicación con Firebase.

Figura 38

Función de interrupción de la señal PWM (> 1.0 um)

```
// ----- SENSOR DHT22 -----
#define DHTPIN 4 // Pin GPIO del ESP32 conectado al DHT22
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// ----- SENSOR MH-Z19 (CO2) -----
SoftwareSerial mhzSerial(16, 17); // RX, TX
unsigned char cmd_get_sensor[] = {0xFF,0x01,0x86,0,0,0,0,0,0};
unsigned char response[9];

// ----- SENSOR PMS5003 (PM2.5) -----
SoftwareSerial pmsSerial(18, 19);

// ----- SENSOR MS-1100 (Formaldehído) -----
#define MS1100_PIN 34 // Pin analógico

// ----- SENSOR MICS6814 (CO) -----
#define MICS6814_PIN 35 // Pin analógico
```

Declaración de sensores y pines

- DHT22: objeto DHT dht (DHTPIN, DHTTYPE); — en tu código DHTPIN está como GPIO 4.
- MH-Z19: módulo CO₂ conectado por UART. Se creó SoftwareSerial mhzSerial (16, 17) (RX=16, TX=17). Para pedir la lectura se envía un paquete de 9 bytes (comando) y se espera una respuesta de 9 bytes.
- PMS5003: sensor de partículas por UART; en el código se creó SoftwareSerial pmsSerial (18, 19). El PMS envía tramas de 32 bytes que empiezan por 0x42 0x4D; el ejemplo solo extrae los bytes para PM2.5.

- MS-1100 (Formaldehído) y MICS6814 (CO): leídos por pines analógicos MS1100_PIN y MICS6814_PIN (ej. GPIO34 y GPIO35). En ESP32 analogRead () devuelve 0–4095 (12 bits por defecto).

Fórmulas a implementar

- *Lowratio (para cada salida PWM)*

$$\text{Lowratio} = \frac{\text{Tiempo}_{\text{LOW}}}{\text{Tiempo}_{\text{Muestreo}}} \times 100$$

- *Conversión a concentración en parts/283mL*

El fabricante proporciona curvas que dan la concentración de partículas en función del Lowratio. El valor obtenido lo llamaremos:

Con1=Concentración(>1um) Con1 = Concentración(>1um)

Con1=Concentración(>1um) Con2=Concentración(>2.5um) Con2 =
Concentración(>2.5um) Con2=Concentración(>2,5um)

Cálculo de PM2.5 en parts/283mL

$$\text{PM2.5}(\text{parts}/283\text{mL}) = \text{Con1} - \text{Con2}$$

Conversión de PM2.5 a µg/m³

Se consideró:

- Radio promedio: $r = 0.875 \mu\text{m} = 0.875 \times 10^{-6} \text{m}$
- Volumen: $V = \frac{4}{3} \pi r^3 = 2.80616 \times 10^{-18} \text{m}^3$
- Densidad: $d = 2 \times 10^{12} \mu\text{g}/\text{m}^3$
- Masa: $m = d \times v = 5.70908 \times 10^{-6} \mu\text{g}$

Finalmente, la conversión queda:

$PM2.5(\mu g/m^3) = PM2.5(\text{parts}/283mL) \times 0.02017$
 $PM2.5(\mu g/m^3) = PM2.5(\text{parts}/283mL) \times 0.02017$

Código del ESP32 principal (AP + STA)

- **Modo 1: WiFi clásico (AP + STA):** El ESP32 actúa como Access Point (AP) y también como cliente (STA), otros ESP32 se conectan a él y todos suben datos a **Firestore**.
- **Modo 2: Red Mesh (PainLessMesh):** Todos los ESP32 forman una malla entre sí y pueden transmitir datos entre nodos sin necesidad de un AP central.

Figura 39

Configuración de parámetros para la red malla

```
#include <WiFi.h>

const char* ssid = "Casino_AP";
const char* password = "12345678";

void setup() {
  Serial.begin(115200);

  // Configurar como Access Point
  WiFi.softAP(ssid, password);
  Serial.println("Access Point iniciado");
  Serial.print("IP del AP: ");
  Serial.println(WiFi.softAPIP());

  // Configurar también como cliente (STA)
  WiFi.begin("TU_ROUTER_SSID", "TU_ROUTER_PASSWORD");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nConectado al router!");
  Serial.print("IP local (STA): ");
  Serial.println(WiFi.localIP());
}
```

Código de los otros ESP32 (clientes STA)

En este modo, un **ESP32 principal** crea la red WiFi (AP) y también se conecta a Internet (STA). Los demás ESP32 se conectan a este AP y usan la conexión para subir sus datos a **Firestore**.

Figura 40*Código fuente*

```

#include <Wifi.h>

const char* ssid = "Casino_AP"; // SSID del ESP32 principal
const char* password = "12345678"; // Password del ESP32 principal

void setup() {
  Serial.begin(115200);

  // Conectar al AP creado por el ESP32 principal
  Wifi.begin(ssid, password);
  while (Wifi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nConectado al ESP32 AP!");
  Serial.print("IP local: ");
  Serial.println(Wifi.localIP());
}

void loop() {
  // Aquí subirías datos a Firebase
}

```

Nota. El código fuente completo está en los anexos

Configuración Red Mesh con PainLessMesh

Aquí todos los nodos se comunican entre sí **sin necesidad de un AP central**. Cada ESP32 puede mandar datos y retransmitir mensajes hacia los demás nodos de la malla.

Figura 41*Código fuente*

```

#include <painlessMesh.h>

#define MESH_PREFIX "CasinoMesh"
#define MESH_PASSWORD "12345678"
#define MESH_PORT 5555

Scheduler usersScheduler;
painlessMesh mesh;

void receivedCallback(uint32_t from, String &msg) {
  Serial.printf("Mensaje recibido de %u: %s\n", from, msg.c_str());
}

void sendMessage() {
  String msg = "Hola desde " + String(mesh.getNodeId());
  mesh.sendBroadcast(msg);
}

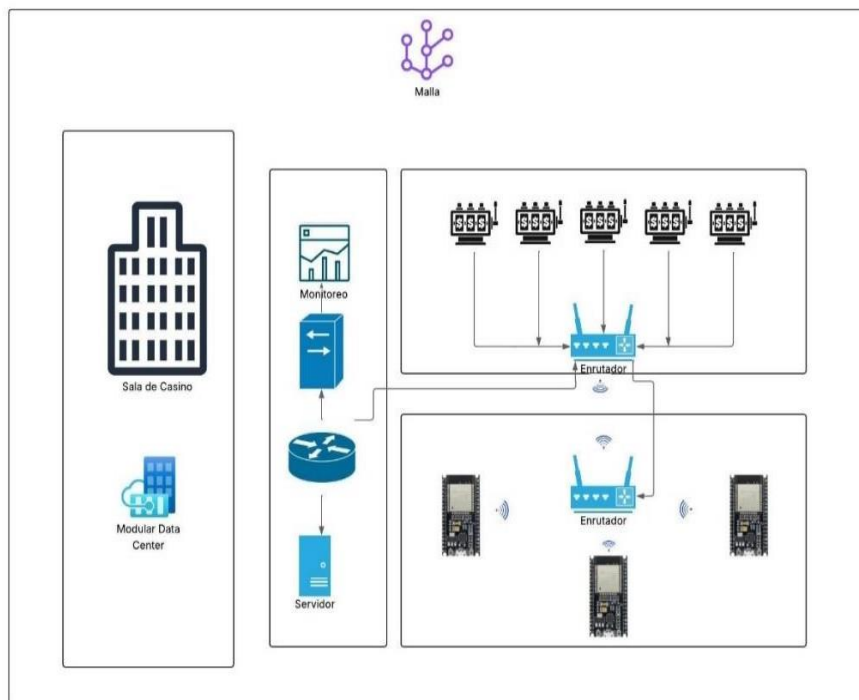
Task taskSendMessage(TASK_SECOND * 5, TASK_FOREVER, &sendMessage);

void setup() {
  Serial.begin(115200);

  mesh.setDebugMsgTypes(ERROR | STARTUP | CONNECTION);
  mesh.init(MESH_PREFIX, MESH_PASSWORD, MESH_PORT, WIFI_AP_STA, 6);
  mesh.onReceive(&receivedCallback);

  usersScheduler.addTask(taskSendMessage);
  taskSendMessage.enable();
}

```

Figura 42*Topología final del proyecto*

Nota. Nótese como se integra las dos redes a través del AP + SPA.

a. Desarrollo del software de monitoreo

En este proyecto se implementó Firebase Realtime Database como plataforma principal de almacenamiento y gestión de datos. Esta elección se debe a que Firebase ofrece una comunicación en tiempo real entre los dispositivos ESP32.

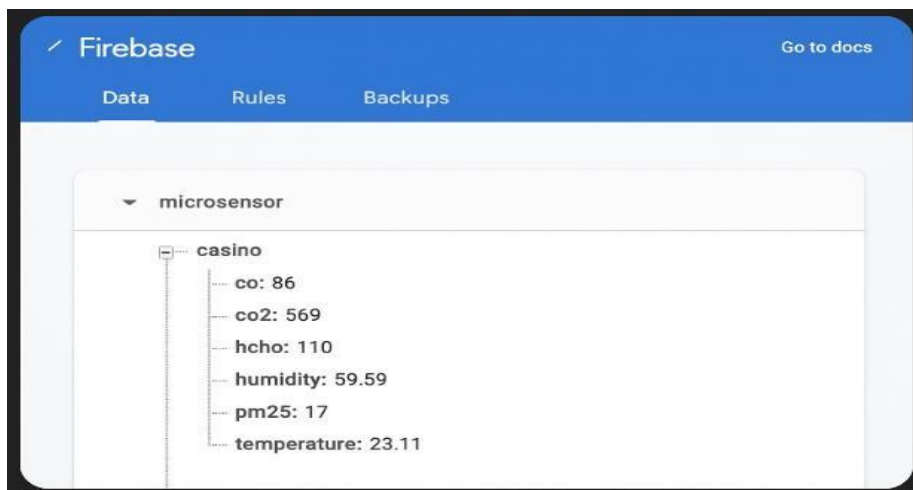
Cada ESP32 está programado para leer diferentes sensores ambientales (CO_2 , $\text{PM}_{2.5}$, temperatura, humedad, formaldehído y gases múltiples). Los valores obtenidos son procesados localmente y luego enviados directamente a Firebase a través del chip WiFi integrado. Para diferenciar las zonas de monitoreo, cada dispositivo utiliza un identificador único (por ejemplo, Sala1 y Sala2), de modo que los registros se almacenan en rutas específicas dentro de la base de datos.

La estructura general en Firebase tiene un nodo raíz denominado Casino, y dentro de este se crean subnodos para cada sala. Dentro de cada sala, se almacenan los parámetros obtenidos de los sensores, tales como se aprecia en el ejemplo siguiente:

Figura 43*Estructura del desarrollo*

```
{
  "Casino": {
    "Sala1": {
      "Temperatura": 25.3,
      "Humedad": 60.1,
      "CO2": 850,
      "PM25_ugm3": 0.54
    },
    "Sala2": {
      "Temperatura": 26.1,
      "Humedad": 55.7,
      "CO2": 920,
      "PM25_ugm3": 0.66
    }
  }
}
```

De esta forma, la información de cada área queda separada y organizada, lo que facilita la consulta y el análisis por parte del software de monitoreo.

Figura 44*Estructura general*

En la aplicación de escritorio, se integró un cliente de Firebase que se encarga de leer periódicamente los valores almacenados en la nube. Estos datos son desplegados en ventanas de monitoreo donde se incluyen cuadros de texto con los valores actuales y gráficas que muestran la evolución de los parámetros de calidad de aire en cada sala

b. Despliegue del prototipo

El montaje y despliegue de la red Malla se realizó en una sala de casino, ubicado en Tacna, Perú. La dirección exacta es Avenida San Martín, Tacna Geográficamente, el casino se encuentra en latitud $18^{\circ} 00' 45''$ Sur de y longitud $70^{\circ} 14' 55''$ Oeste.

Figura 45.

Ubicación de la zona de despliegue de la red malla



Nota. Fuente de Google Earth

Esta red de sensores y nodos ESP32, desplegada en estas instalaciones, permitió establecer una arquitectura robusta de monitoreo ambiental en red malla, aprovechando la conectividad distribuida del entorno del casino para garantizar una transmisión eficiente de datos hasta Firebase. Para la ubicación exacta de los nodos finales (sensores) dentro de la sala de fumadores, se consideró como criterios, la circulación del aire y el comportamiento físico de los gases presentes en el ambiente.

En este tipo de espacio cerrado, el humo del tabaco contiene partículas sólidas (PM2.5, PM10) y gases contaminantes como el CO y el CO₂. Dado que el CO₂ tiene una densidad ligeramente mayor que el aire, tiende a acumularse en niveles bajos, mientras que las partículas finas y el humo en suspensión se distribuyen con rapidez en toda la sala debido a las corrientes de aire generadas por el ingreso de personas y el sistema de ventilación.

El mayor flujo de aire se concentra en la puerta de acceso y en los extractores de ventilación, lo que provoca que en las esquinas superiores de la sala se generen turbulencias que acumulan contaminantes. Por esta razón, se consideró óptimo ubicar los nodos de monitoreo a una altura aproximada de 2 metros sobre el nivel del suelo, en las esquinas superiores de la sala de fumadores.

De esta forma, se logra un registro más representativo de la concentración de material particulado y gases contaminantes en las zonas donde suelen permanecer los fumadores, garantizando una medición precisa y acorde al comportamiento dinámico de los contaminantes en este tipo de ambientes. Ver figura 45.

Figura 46

Ubicación de la sala de fumadores

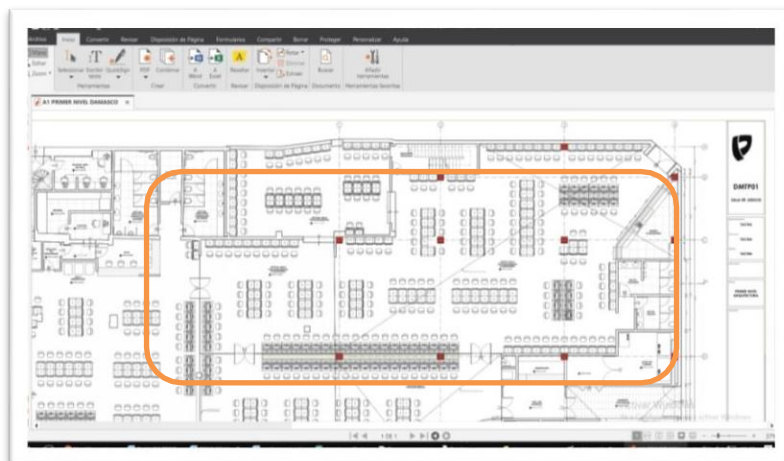


Figura 47

Vista de la ubicación del AP (Access point) dentro de la sala de fumadores



3.3. Materiales e Instrumentos.

3.3.1. Materiales

- *Estaño*: esencial para la soldadura de componentes a la placa y la unión de

terminales desde el ESP32 hacia los sensores.

- *Alicate de corte fino.*
- *Silicona.*
- *Caja de paso cuadrada:* Diseñada por impresora 3D acorde a las dimensiones de los componentes.
- *Interruptor On-Off:* utilizado para la interrupción del funcionamiento del dispositivo IoT.
- *Cableado:* Pequeños Cables para la unión de componentes y durante el proceso de prueba con el protoboard.
- *Vidrio templado:* utilizado para la realizar la simulación de un ambiente de fumadores.

3.3.2. Instrumentos

- *Multímetro:* Equipo usado para medición de voltajes, continuidad y resistencia
- *Medidor de calidad de aire:* Equipo usado para tener un punto de referencia respecto a los parámetros de calidad de aire obtenidos del prototipo IoT.
- *Pistola para medir temperatura:* Necesaria para la medición de temperatura dentro del cubículo de vidrio.
- *Cautín 60watts:* Para la unión de los puntos de conexión entre la placa y los componentes (sensores y ESP32).
- *Cable de conexión USB 3.0:* Usado para el envío de información del módulo ESP32 hacia la PC de gestión.
- *Laptop personal:* Usado para la implementación del software y programación del módulo ESP32, Marca Asus Tuf f15.
- *Placa de circuito impreso:* Diseñado por la máquina de elaboración de placas bajo, se tomó en cuenta el diseño del circuito impreso para su elaboración.
- *Máquina de elaboración de placas:* Alquilado para la elaboración de la placa.
- *Impresora 3D:* Alquilado para la elaboración del case del dispositivo.

3.4. Población y/o muestra de estudio.

3.4.1. Muestra de estudio

La muestra de estudio o la unidad de análisis propiamente dicho, se refiere al Prototipo IoT para la medición y monitoreo de la calidad de aire en las salas de fumadores de un casino.

3.5. Operacionalización de variables

En la Tabla 15 se detalla la operacionalización de variables de nuestro trabajo de investigación.

Tabla 15

Operacionalización de variables

Variable	Definición Conceptual	Dimensiones	Indicadores	Técnicas y métodos
Variable 1: Prototipo IoT para medición y monitoreo	Un sistema IoT (Internet de las Cosas) es una red de dispositivos físicos equipados con sensores y software que les permiten conectarse a Internet y recopilar e intercambiar datos.	- Diseño del prototipo	<ul style="list-style-type: none"> • Especificación de sensores integrados (tipo y cantidad) • Capacidad de transmisión (bps) • Almacenamiento de datos (Mbyte) 	<ul style="list-style-type: none"> • Se aplicará metodología ágil (Scrum) para organizar y validar • iterativamente las etapas de desarrollo y pruebas del prototipo.
Variable 2: Calidad de aire en salas de fumadores	La concentración de contaminantes del humo del tabaco y otros agentes, medidos por el Índice de Calidad del Aire (ICA), el cual evalúa la seguridad y los efectos en la salud.	<ul style="list-style-type: none"> - Contaminantes medidos - Condiciones ambientales 	<ul style="list-style-type: none"> • Nivel de concentración de CO (ppm) • Nivel de concentración de CO₂ (ppm) • Nivel de concentración de PM_{2.5} (µg/m³) • Temperatura ambiente (°C) • Humedad relativa (%) 	<ul style="list-style-type: none"> • Método observacional con registro automático a través del prototipo IoT para captar en tiempo real los niveles de contaminantes y condiciones ambientales

3.6. Procesamiento y análisis de datos

3.6.1. Base de datos

En este proyecto se llevó a cabo el procesamiento y análisis de los datos obtenidos a través de los sensores conectados al ESP32, destinados a medir parámetros de calidad del aire en ambientes cerrados como la sala de fumadores de un casino.

El flujo completo incluye la captura de señales, su procesamiento matemático, la transmisión de los datos mediante la comunicación WiFi hacia Firebase Realtime Database, y finalmente la visualización e interpretación dentro del software de monitoreo y control.

3.6.2. Etapas del procesamiento

a. Captura de señales

Los sensores conectados al ESP32 (MH-Z19, PMS5003, DHT22, MS-1100, MICS6814, entre otros) envían datos analógicos o digitales que son interpretados por el microcontrolador. En el caso del sensor DSM501A, se obtiene la señal PWM correspondiente a la concentración de partículas.

b. Cálculo del Lowratio

Se mide el tiempo en que la señal permanece en bajo (LOW) durante un intervalo de muestreo. El porcentaje de este tiempo con respecto al tiempo total se denomina 'Lowratio'. Este valor es fundamental para estimar la concentración de partículas suspendidas en el aire.

c. Conversión a concentración de partículas

El Lowratio se convierte a unidades de concentración (parts/283mL) utilizando la curva de calibración proporcionada por el fabricante. Posteriormente, se realiza la conversión a unidades estándar ($\mu\text{g}/\text{m}^3$). La fórmula aplicada es la siguiente:

$$\text{PM}_{2.5} (\mu\text{g}/\text{m}^3) = \text{PM}_{2.5} (\text{parts}/283\text{mL}) \times 0.02017$$

d. Transmisión de datos

Una vez procesados, los datos son enviados a Firebase Realtime Database utilizando el chip WiFi integrado en el ESP32. Cada dispositivo reporta sus valores a una ruta específica dentro de la base de datos, permitiendo diferenciar las distintas zonas de monitoreo (ejemplo: Sala 1, Sala 2).

e. Almacenamiento y estructuración

Los datos son almacenados en formato JSON dentro de Firebase, estructurados de manera jerárquica. Esto permite un acceso organizado y en tiempo real por parte del software de monitoreo

f. Visualización y análisis

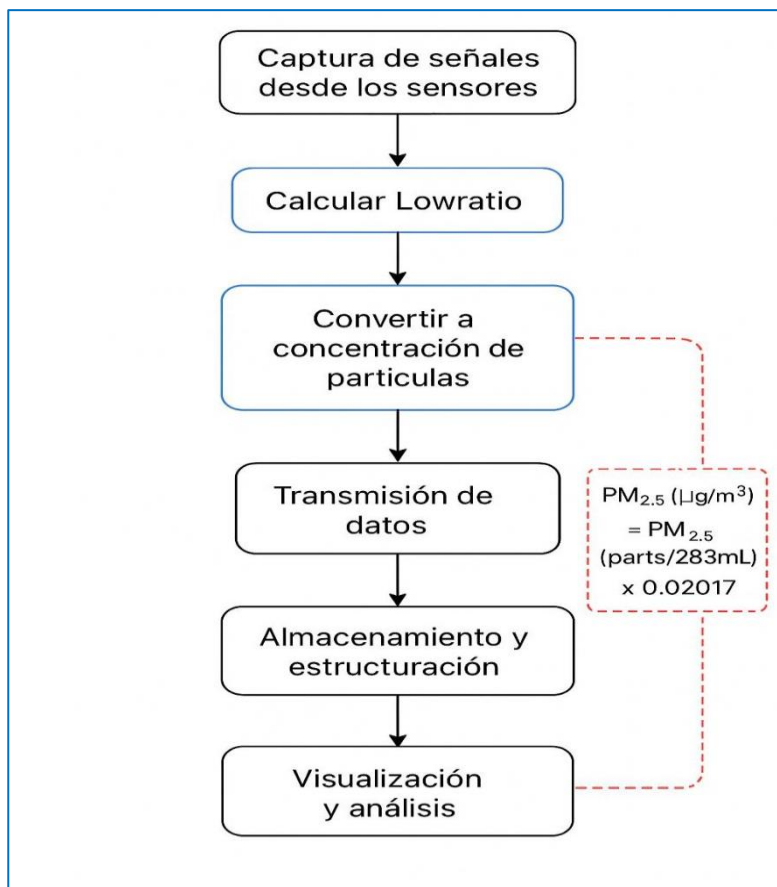
En la aplicación de escritorio desarrollada en Visual Studio con WinForms, los datos se representan en gráficas y cuadros de texto, lo que permite analizar la evolución de parámetros como CO₂, PM_{2.5}, temperatura y humedad, comparándolos con los valores máximos permisibles establecidos por el MINAM.

3.6.3. Diagrama de flujo del procesamiento de datos

Para el procesamiento de datos se estableció el siguiente diagrama de flujo, tal como se muestra en la Figura 47.

Figura 48

Diagrama de flujo de procesamiento de datos



CAPÍTULO IV: RESULTADOS

4.1. Medidor de calidad de aire

Como referencia, se utilizó el equipo de medición de la calidad del aire en la sala del casino. Las muestras y lecturas de los parámetros se muestran en la siguiente tabla:

Tabla 16

Lectura del equipo medidor de calidad de aire

Compuesto	Tiempo de muestreo				
	5min.	20 min.	45 min.	1 hora	8 horas
Monóxido de carbono	0 ppm	0 ppm	0 ppm	0 ppm	0 ppm
Dióxido de carbono	442 ppm	587 ppm	692 ppm	924 ppm	1831 ppm
Formaldehido	0,0014 mg/m ³	0,0012 mg/m ³	0,0016 mg/m ³	0,023 mg/m ³	0,069 mg/m ³
PM2.5	12 ug/m ³	14 ug/m ³	13 ug/m ³	12 ug/m ³	16 ug/m ³
Temperatura	23° C	22° C	23°C	20° C	17° C
Humedad	43%	54%	60%	75%	89%

Nota: En el caso del valor de la temperatura y humedad se usó otro dispositivo móvil para las lecturas en la tabla anterior.

4.2. Prototipo IoT de medición de calidad de aire

Una vez calibrado el equipo según las lecturas del medidor de calidad del aire, se procedió a tomar las mediciones de la misma forma, obteniendo los siguientes resultados:

Tabla 17*Lectura de prototipo IoT para la medición de calidad de aire*

Compuesto	Tiempo de muestreo				
	5min.	20 min.	45 min.	1 hora	8 horas
Monóxido de carbono	4,4 ppm	4,3 ppm	4,4 ppm	6,2 ppm	9,4 ppm
Dióxido de carbono	1320 ppm	1708 ppm	2135 ppm	5147 ppm	4582 ppm
Formaldehido	0	0	0	0	0
PM2.5	12 ug/m3	13 ug/m3	15 ug/m3	12 ug/m3	17 ug/m3
Temperatura	24° c	23° C	21°C	20° C	19° C
Humedad	66%	68%	68%	71%	76%

Nota: lecturas realizadas después de 1 hora de curación desde el encendido del prototipo.

4.3. Parámetros a medir de la calidad de aire

4.3.1. Monóxido de carbono (CO)

El monóxido de carbono (CO) se expresa en ppm (partículas por millón) tanto en el prototipo IoT como en el medidor de calidad del aire. Durante las etapas iniciales de prueba, el CO no se presentó en ambientes con contaminación elevada, por lo que las mediciones de muestra fueron relativamente bajas en comparación con las demás unidades.

Los resultados de las pruebas realizadas en ambientes con alta contaminación se muestran en la Tabla 17

Tabla 18*Lecturas desde el medidor y prototipo IoT para el CO*

Monóxido De Carbono			
Tiempo de muestra (minutos)	Medidor de calidad de aire (ppm)	Prototipo IoT (ppm)	Valor permitido por la OMS (ppm)
5	3.8	4.4	6400
20	7.9	8.6	1600
45	15.2	18.4	800
60	85	91	400
480	32	43	20

Análisis de los resultados

Se puede observar que luego de 8 horas (480 minutos), el medidor de calidad obtuvo una lectura de 32 ppm y el prototipo 43 ppm. Al comparar los valores con los límites permitidos por la OMS (20 ppm) existe una elevada contaminación en la sala de fumadores del casino.

4.3.2. Dióxido de carbono (CO₂)

La Tabla 18 muestra las lecturas obtenidas de CO₂, tanto en el medidor de calidad de aire y el prototipo desarrollado en esta tesis.

Tabla 19*Lecturas desde el medidor y prototipo IoT para el CO₂*

Dioxido De Carbono			
Tiempo de muestra (minutos)	Medidor de calidad de aire (ppm)	Prototipo IoT (ppm)	Valor permitido por la OMS (ppm)
5	4370	5141	2100
20	3298	3829	1400
45	1387	1243	1100
60	852	748	1200
480	706	613	800

Análisis de los resultados

La Tabla 18 muestra la diferencia de resultados en un lapso de trabajo de 5 a 8 horas donde el ambiente de la sala de fumadores, donde el CO presenta una variación mucho más alta, esto debido a los compuestos producidos por el tabaco en el ambiente, la poca ventilación y la aglomeración de personas que dificulta la ventilación adecuada.

Al igual que el CO, el dióxido de carbono también se muestran en ppm, y a diferencia del CO las lecturas obtenidas de ambos dispositivos son diferentes, esto se debe a que la lectura de los sensores en el prototipo IoT es mucho más preciso que los sensores del medidor de calidad de aire.

Para el MHZ19 (sensor de CO₂), las lecturas debieron tomarse luego de conectar el prototipo IoT por 24 horas, en este punto es donde las lecturas se tomaron ya de forma más precisa para la elaboración de tablas comparativas junto al equipo medidor de calidad de aire.

4.3.3. Formaldehído (HCHO)

El formaldehído al ser un gas incoloro, inflamable y de fuerte olor no se presentaron en gran cantidad durante las pruebas de lectura iniciales para el prototipo IoT, tampoco en las lecturas hechas en el medidor de calidad de aire para obtener una lectura base. La Tabla 19 muestra los valores del gas CH₂O (formaldehído) tanto con el medidor de calidad y de nuestro prototipo.

Tabla 20

Lecturas desde el medidor y prototipo IoT para el formaldehído

Formaldehído CH ₂ O						
Tiempo muestra (minutos)	deMedidor de calidad de aire (ppm)	de Medidor de calidad de aire (mg/m ³)	Prototipo IoT (ppm)	Prototipo IoT (mg/m ³)	Valor permitido por la OMS (mg/m ³)	
5	0.016	0.013024	0.23	0.18722	15	
20	0.021	0.017094	0.9	0.7326	2.5	
45	0.023	0.018722	1.53	1.24542	1.5	
60	0.038	0.030932	1.9	1.5466	1	
480	0.04	0.03256	2.4	1.9536	0.75	

Nota: se realizó la conversión de ppm a mg/m³ en el caso del prototipo IoT ya que la programación con la que viene de fábrica arroja valores digitales y no analógicos.

Análisis de los resultados

Durante las pruebas reales se obtuvieron lecturas de CH₂O muy bajas en los primeros minutos, tanto en el medidor de calidad del aire y nuestro prototipo. Conforme fue transcurriendo el tiempo de labores en la sala de fumadores del casino, los valores llegaron a sobrepasar los valores permitidos por los estándares de salud de la OMS, lo cual permitió que el ventilador se activara para reducir los niveles de CH₂O.

4.3.4. Material particulado 2.5 (PM2.5)

La Tabla 20 muestra los valores del material particulado 2.5 (PM2.5) tanto con el medidor de calidad y de nuestro prototipo

Tabla 21

Lecturas desde el medidor y el prototipo IoT para PM2.5

Material Particulado (PM2.5)			
Tiempo de muestra (minutos)	Medidor de calidad de aire (ug/m3)	Prototipo IoT (ug/m3)	Valor permitido por la OMS (ug/m3)
5	485	39	>110
20	123	28	56
45	47	21	31
60	16	14	16
480	4	3	1

Nota: Las lecturas tomadas desde el medidor de calidad de aire se hicieron sin las 24 horas de curación para los sensores, los resultados fueron estabilizados progresivamente.

Análisis de los resultados

En el caso del material particulado solo se presentaron lecturas durante las pruebas de medición de calidad de aire en las salas de casino, ya que en ambientes naturalmente aspirados o con ventilación propiamente adecuada no llegan al umbral máximo permitido por los estándares de salud.

4.3.5. Temperatura (C°)

Para la comparativa de la temperatura no se tienen lecturas del medidor de calidad de aire ya que este dispositivo solo cuenta con 6 parámetros de lectura por cuestión de

modelo y fabricación, solo se tomaron muestras con un medidor infrarojo de temperatura, las lecturas del prototipo si fueron muy cercanas a las del medidor infrarojo, comprobando la precision de los sensores en el prototipo IoT.

4.3.6. Humedad (Hr %)

La humedad al ser un parámetro dentro de la calidad de aire, se tomo en cuenta para los ambientes de la sala de fumadores de un casino, al igual que la temperatura presentó lecturas desde el prototipo IoT cercanas a las lecturas del medidor de humedad movil, estas lecturas son importantes para las comparaciones de los resultados finales y que puedan demostrar que los niveles de la calidad de aire son óptimos o perjudiciales para la salud del personal dentro de las salas de fumadores.

Figura 49

Resultados de las 10 primeras muestras de cada sensor en tiempo real

1	s_fumadores.ino									
2	Tiempo	Temperatura	Humedad	CO ₂	PM2.5	HCHO	CO	Lowratio	PM2.5 (ppv/0.01cf)	PM2.5 (µg/m ³)
4	2025-09-09 19:00:00	25.3°C	58.1 %	1245	87.2	0.10	3.5	2.41 %	145	35.2
5	2025-09-09 19:10:00	25.6°C	57.9 %	1302	92.5	0.11	3.7	2.58 %	153	37.0
6	2025-09-09 19:20:00	25.8°C	57.5 %	1360	95.8	0.12	3.9	2.65 %	160	38.6
7	2025-09-09 19:30:00	26.0°C	57.2 %	1425	98.3	0.13	4.0	2.71 %	162	39.0
8	2025-09-09 19:40:00	26.3°C	57.0 %	1488	101.2	0.13	4.2	2.79 %	167	40.1
9	2025-09-09 19:50:00	26.4°C	56.7 %	1535	104.5	0.14	4.3	2.85 %	172	41.4
10	2025-09-09 20:00:00	26.6°C	56.4 %	1602	107.9	0.15	4.4	2.91 %	176	42.6
11	2025-09-09 20:10:00	26.8°C	56.2 %	1655	110.1	0.15	4.6	2.97 %	179	43.4
12	2025-09-09 20:20:00	27.0°C	56.0 %	1701	112.8	0.16	4.7	3.04 %	183	44.5
13	2025-09-09 20:30:00	27.2°C	55.8 %	1750	115.3	0.16	4.9	3.10 %	187	45.7

CAPÍTULO V: DISCUSIONES

Durante las pruebas realizadas, nos encontramos con problemas en la implementación de los dispositivos y sensores. Las lecturas iniciales con los sensores del tipo MQ no fueron los esperados, cada sensor arrojaba resultados no precisos, es por ello que optamos por el uso de los sensores de más precisión mencionados durante el presente proyecto.

En la revisión de los resultados obtenidos se comprobó que el prototipo desarrollado en esta investigación presenta mejoras significativas respecto al antecedente internacional de Guatapi y García (2022), quienes implementaron un sistema de medición y purificación de gases basado en Arduino y la plataforma UBIDOTS IoT. La principal optimización se observa en el ámbito del hardware y del procesamiento, ya que mientras el sistema de Guatapi y García utilizaba un microcontrolador de 8 bits con arquitectura básica y una frecuencia de reloj de 16 MHz, el presente proyecto incorpora un SoC ESP32 con arquitectura dual-core de 32 bits y un procesamiento a 240 MHz, lo que incrementa la capacidad de cómputo, la velocidad de respuesta y la eficiencia en la ejecución simultánea de tareas. Asimismo, a diferencia del antecedente citado, el ESP32 integra de forma nativa módulos Wi-Fi y Bluetooth, lo que elimina la necesidad de componentes adicionales y mejora la estabilidad, la seguridad y el rendimiento de la conectividad en el sistema IoT.

Durante la implementación del Sistema IoT en el casino, se comprobó que la red basada en módulos ESP32, logro cubrir las zonas de interés (salas de fumadores) sin necesidad de instalar puntos de acceso adicionales, la latencia promedio registrada en la transmisión de datos desde el prototipo IoT hasta el servidor central fue de 240 ms en un máximo de 300 ms en condiciones de mayor tráfico. Dichos valores se encuentran dentro de los rangos aceptables para aplicaciones de monitoreo, donde el requisito crítico es el muestreo de datos en tiempo real.

Como resultado se obtuvo valores proporcionados por los estándares de salud respecto a la calidad de aire establecidos por la OMS, estos valores pudieron ser verificados en tiempo real con variaciones en sus lecturas propias de los sensores del prototipo IoT, las variaciones tienen un porcentaje de error dentro de lo normal respecto a los dispositivos de lectura para cada parámetro, la calibración de estas se realizó en el mismo lenguaje de programación para cada parámetro, optimizando los resultados y la calidad de la lectura similar a la de un medidor de precisión y con una estructura más detallada que se puede visualizar en el servidor firebase.

CONCLUSIONES

Se llegaron a las siguientes conclusiones:

Se comprobó que la lectura de los parámetros críticos a medir de la calidad de aire se puede visualizar por medio de sensores de precisión con bajo retardo en sus lecturas.

Se comprobó que es posible obtener una lectura en tiempo real de los parámetros que conforman la calidad de aire, lo que permite resguardar la salud de las personas al interior de establecimientos como lo son la sala de fumadores.

Los sensores empleados en el prototipo como el MHZ19 para el CO₂, PMS5003 para PM_{2.5} y DHT22 para temperatura y humedad fueron acertados, dado que proporcionaron la información de manera confiable de los contaminantes más relevantes en los ambientes cerrados de la sala de fumadores del casino, contribuyendo al monitoreo en beneficio del usuario.

Se comprobó que para algunos sensores como el MICS6814 para el CO, requieren una fórmula para calibrar la entrega de su valor analógico, ya que en su hoja de datos presenta una tabla con valores aproximados a los digitales y en base a ello se puede calibrar un resultado óptimo en el prototipo IoT.

Se compro la respuesta de las lecturas hechas por el prototipo IoT respecto a la salida hacia el relee incorporado en la placa, dicho relee presente tiene solo una función, la de activar el extractor luego de un tiempo, el cual están basado en los tiempos en la cual una persona puede estar expuesta a determinado nivel de contaminación en el aire según la OMS.

Finalmente, en las pruebas realizadas en campo, se comprobó que los parámetros de calidad de aire medidos en la sala de fumadores del casino, no se encuentran dentro de los límites permitidos en los estándares de calidad ambiental establecidos por la Organización Mundial de la Salud (OMS), evidenciando concentraciones de CO₂ superiores a 2500 ppm y de PM_{2.5} por encima de 75 µg/m³ en periodos de mayor concurrencia, estos valores se consideran no permitidos para ambientes cerrados, evidenciando la necesidad de la implementación del presente proyecto IoT para el control de la calidad de aire.

RECOMENDACIONES

Se presentaron las siguientes recomendaciones:

Se recomienda la implementación del sistema de ventilación y filtración de partículas, ante los niveles no permitidos de CO₂ y Pm_{2.5} se recomienda instalar Sistema de ventilación mecánica con filtros, integrados al Sistema de monitoreo para activarse de forma automática cuando se superen los umbrales definidos.

Se recomienda a futuro establecer un monitoreo con mensajes de alarma para obtener un aviso anticipado en tiempo real, el cual podrá mejorar las acciones a tomar y prevenir cualquier complicación que afecte la salud de los usuarios en las salas de fumadores del casino.

Se recomienda indagar la aplicación de otras tecnologías para la implementación del software ya que la plataforma creada durante el proyecto es gratuita y no es de uso exclusivo para la sala de control.

Las muestras que el presente proyecto registro corresponden a sensores de alto costo, que, si bien han sido configurados por defecto por parte de los fabricantes, se recomienda la calibración instrumentos de medida correspondientes a cada parámetro según la normativa para la calidad de aire.

REFERENCIAS BIBLIOGRÁFICAS

- AirQ (s.f) Información sobre los valores límite de gases, partículas y contaminantes en el aire que respiramos. <https://es.air-q.com/grenzwerte>
- Air Quality detector. Datasheet del equipo medidor de calidad de aire
<https://5.imimg.com/data5/SELLER/Doc/2025/12/565524411/BB/QR/JN/391686/air-quality-detector-zn-2co7-1.pdf>
- Asthon, K. (2009). Eso del 'Internet de las Cosas'. RFID Journal.
<https://www.rfidjournal.com/expert-views/that-internet-of-things-thing/73881/>
- Castillo, A. (2020). Cable par trenzado — Características, construcción, tipos y categorías, Profesional Review.
<https://www.profesionalreview.com/2020/09/12/cable-par-trenzado-caracteristicas/>
- C & D Tecnologia (2025). Sensor de formaldehido modelo MS-1100.
<https://cdtecnologia.net/sensores/1328-sensor-de-gas-formaldehido-benceno-tolueno-cjmcu-1100-ms-1100-cov.html>
- Chacón, J., & Cardona, M. (2023). El internet de las cosas (IoT) para el monitoreo de la calidad del aire en salas de fumadores de casinos. Revista de Tecnología e Innovación, Pag. 15-28. <https://repositorio.espe.edu.ec/handle/21000/36761>
- Guatapi, A. y García, E. (2022). Diseño e implementación de un prototipo para un sistema de medición, análisis y purificador de gases contaminantes en el aire utilizando Arduino y UBIDOTS IoT. [Tesis de Título, Universidad Politécnica Salesiana], Guayaquil, Ecuador.
<https://dspace.ups.edu.ec/bitstream/123456789/24088/1/UPS-GT004189.pdf>
- Klepeis, N. E., Ott, W. R., Switzer, P., & et al. (2016). Measuring Indoor Air Quality and Engaging California Indian Casinos to Reduce Secondhand Smoke Exposure. International Journal of Environmental Research and Public Health, 13(1), 143.
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4730534/>
- Kurose, J. y Ross, K. (2010). Redes de computadores: Capítulo 6: Redes inalámbricas y móviles. Pearson Educación, 2010.
<https://www.cartagena99.com/recursos/alumnos/apuntes/Tema%203-Redes%20inalambricas%20y%20moviles.pdf>

Medina de la Cruz, M. (2020). Diseño e implementación de un dispositivo de Internet de las cosas para la medición y monitoreo de la calidad de aire en zonas urbanas. [Tesis de Título, Universidad de Ciencias y Humanidades], Lima, Perú.

https://repositorio.uch.edu.pe/bitstream/handle/20.500.12872/506/Medina_MR_tesis_ingenieria_electronica_2020.pdf?sequence=1&isAllowed=y

Melnyk, I. (2016). MH-Z19 CO2 meter.

https://ihormelnyk.com/mh-z19_co2_meter?srsId=AfmBOopd_QG2wVGboVXCKp45LJsHi_cwR1WM_uxqEFT3Y4kUW0Hz43q1Q

Mendoza, R. y Cordero, V. (2022). Diseño de un sistema IoT para el monitoreo remoto de la calidad del aire en ambientes interiores de la Facultad de Ingeniería de la Universidad Privada de Tacna. [Tesis de Título, Universidad Privada de Tacna], Tacna Perú.

https://repositorio.upt.edu.pe/handle/20.500.12969/2630?utm_source=chatgpt.com

MICRODESYS (2023). Microcontrolador Esp32.

<https://microdesys.es/docs/microcontrolador-esp32/>

Ministerio del Ambiente (MINAM). (2019). Decreto Supremo N° 010-2019- MINAM: Norma técnica de gestión ambiental para el manejo de residuos sólidos en las actividades de minería y metalmecánica.

Othman, H., Azari, R. y Guimarães, T. (2024). Monitoreo económico de la calidad del aire interior basado en IoT. *Tecnología Arquitectura + Diseño*, 8 (2), 250–270.

<https://doi.org/10.1080/24751448.2024.2405403>

Panagiotis I., Grammatikis, R., Sarigiannidis P. y Moscholios L. (2019). Securing the Internet of Things: Challenges, threats and solutions, *Internet of Things*, Volume 5, 2019, pp. 41-70, ISSN 2542-6605

<https://www.sciencedirect.com/science/article/pii/S2542660518301161>

Quishpe, J. P., & Toapanta, S. M. (2023). Diseño e implementación de un prototipo para el monitoreo de la calidad del aire en la sala de fumadores del Casino de la Tripulación de la Armada. <https://repositorio.espe.edu.ec/handle/21000/36761>

Risteska, B. y Trivodaliev, K. (2016). Una revisión del Internet de las cosas para el hogar inteligente: desafíos y soluciones. *Journal of Cleaner Production*, Volume 140,

Part 3, 2017, Pages 1454-1464,ISSN 0959-6526,

<https://doi.org/10.1016/j.jclepro.2016.10.006>.<https://www.sciencedirect.com/science/article/pii/S095965261631589X?via%3Dihub>

SAISAC MECATRÓNICA (2024). Sensor de temperatura y humedad DHT22 con cable.

<https://mecatronica.saisac.pe/producto/dht22-con-cable/>

Unit Electronics (2023). Sensor de material particulado PMS5003.

<https://uelectronics.com/producto/pms5003-sensor-de-calidad-de-aire-pm25/?srsrtid=AfmBOorPNWV8eUav5xG7nq4s0RteTSIIJUyB3rbU31nLgipg-PLwO04J>

UTEL (2020). Medios de Transmisión.

<https://apps.utel.edu.mx/recursos/files/r161r/w24365w/Medios%20de%20transmision%202.pdf>

Vermesan, O. and Friess, P. Editors (2013) Internet of Things— From Research and Innovation to Market Deployment. River Publishers Series in Communication, Denmark.

https://www.riverpublishers.com/pdf/ebook/RP_E9788793102958.pdf

York et al. (2010). A baseline evaluation of casino air quality after enactment of a clean indoor air act. PubMed.

<https://pubmed.ncbi.nlm.nih.gov/20433670/>

Zevallos, J. (2024). Monitoreo de la calidad de aire interior basado en arquitectura IOT. [Tesis de Título, Universidad de Lima], Lima, Perú.

<https://hdl.handle.net/20.500.12724/20543>

Zhou, M., Abdulghani, A. M., Imran, M. A. and Abbasi, Q. H. (2020) Internet of Things (IoT) Enabled Smart Indoor Air Quality Monitoring System. In: 2020 International Conference on Computing, Networks and Internet of Things (CNIOT 2020), , Sanya China, 24-26 Apr 2020, pp. 89-93. ISBN 9781450377713 (doi:10.1145/3398329.3398342).

ANEXOS

ANEXO 1: MATRIZ DE CONSISTENCIA

Problema	Objetivos	Hipótesis	Variables de estudio	Dimensión e Indicador	Metodología
<p>Problema general</p> <p>¿Cómo diseñar e implementar un prototipo IoT para la medición y monitoreo continuo de la calidad del aire en las salas de fumadores de un casino?</p>	<p>Objetivo general</p> <p>Diseñar e implementar un prototipo IoT para la medición y monitoreo de calidad de aire en las salas de fumadores de un casino.</p>	No aplica	<p>Variable 1</p> <p>Prototipo IoT para medición y monitoreo</p>	<p>Diseño del prototipo</p> <p><u>Indicadores:</u></p> <ul style="list-style-type: none"> - Especificación de sensores integrados (tipo y cantidad). - Capacidad de transmisión (bps). - Almacenamiento de datos (Mbyte) 	<p>Tipo de investigación:</p> <ul style="list-style-type: none"> - Investigación aplicada <p>Nivel de investigación:</p> <ul style="list-style-type: none"> - Explicativo <p>Diseño de investigación</p> <ul style="list-style-type: none"> - Experimentales <p>Técnicas y métodos</p> <p><u>Variable 1</u></p> <ul style="list-style-type: none"> - Se aplica metodología ágil (Scrum) para organizar y validar iterativamente las etapas de desarrollo y pruebas del prototipo.
<p>Problemas específicos</p> <p>a. ¿Cuáles son los parámetros ambientales críticos que afectan la calidad del aire en las salas de fumadores de un casino?</p> <p>b. ¿Qué dispositivos y sensores IoT permiten una medición precisa y en tiempo real de la calidad del aire en ambientes cerrados?</p> <p>c. ¿Cómo se puede integrar un sistema IoT de monitoreo en tiempo real con mecanismos de control ambiental para mejorar las condiciones</p>	<p>Objetivos específicos</p> <p>a. Determinar los parámetros ambientales críticos que afectan la calidad del aire en las salas de fumadores de un casino.</p> <p>b. Determinar los dispositivos y sensores IoT permiten una medición precisa y en tiempo real de la calidad del aire en ambientes cerrados.</p> <p>c. Integrar un sistema IoT de monitoreo en tiempo real con mecanismos de control ambiental para mejorar las condiciones en las salas de fumadores.</p>	No aplica	<p>Variable 2</p> <p>Calidad de aire en salas de fumadores</p>	<p>Contaminantes medidos</p> <p><u>Indicadores:</u></p> <ul style="list-style-type: none"> - Nivel de concentración de CO (ppm) - Nivel de concentración de CO₂ (ppm) - Nivel de concentración de PM_{2.5} (µg/m³) <p>Condiciones ambientales</p> <p><u>Indicadores</u></p> <ul style="list-style-type: none"> - Temperatura ambiente (°C) - Humedad relativa (%) 	<p><u>Variable 2</u></p> <ul style="list-style-type: none"> - Método observacional con registro automático a través del prototipo IoT para captar en tiempo real los niveles de contaminantes y condiciones ambientales.

ANEXO 2: CODIGO DE PROGRAMACION PARA EL ESP32 EN ARDUINO 1.8.15

```
1  ///////////////////////////////////////////////////////////////////
2  #if defined(ESP32)
3    #include <WiFi.h>
4  #elif defined(ESP8266)
5    #include <ESP8266WiFi.h>
6  #endif
7
8  #include <Firebase_ESP_Client.h>
9  // Insertar credenciales de red
10 #define WIFI_SSID "tesiscalidadcasino"//define WIFI_SSID
11 #define WIFI_PASSWORD "01020304"//define WIFI_PASSWORD
12 // Proporcionar la informacion del proceso de generacion del token.
13 #include "addons/TokenHelper.h"
14 // Proporcionar la informacion de impresion de la carga util de RTDb.
15 #include "addons/RTDBHelper.h"
16 // Insertar la API Key del proyecto Firebase
17 #define API_KEY "AlzaSyAK7qXSHvmCzvuzXG7WjqSRo1ovu19tyhY"// //
18 // Insertar la URL de la base de datos RTDB
19 #define DATABASE_URL "https://tesis-calidad-de-aire-4919e-default-rtdb.
20 firebaseio.com/"//
21 https://tesis-calidad-de-aire-4919e-default-rtdb.firebaseio.com/ // -m/
22 // Definir objeto de datos de Firebase
23 FirebaseDatabase fbdo;
24 // Definir autenticacion de Firebase
25 FirebaseAuth auth;
26 // Definir configuracion de Firebase
27 FirebaseConfig config;
28 // Variable booleana para el estado de registro
29 bool signupOK = false;
30 ///////////////////////////////////////////////////////////////////
31 ///////////////////////////////////////////////////////////////////MICS6814/////////////////////////////////////////////////////////////////
32 #include <math.h>
33 #include <Arduino.h>
34 float Co;
35 char CoString[10];
36 // Enum for the sensor channels
37 enum channel
38 {
39     CH_NH3, CH_RED, CH_OX
40 };
41 typedef enum channel channel_t;
42 // Enum for proper gas declaration
43 enum gas
44 {
```

```

44   CO, NO2, NH3, C3H8, C4H10, CH4, H2, C2H5OH
45 };
46 typedef enum gas gas_t;
47
48 #define COPIN 34//32//A3
49
50 uint16_t REDbaseR;
51
52 uint16_t getResistance(channel_t channel)
53 {
54     unsigned long rs = 0;
55     int counter = 0;
56
57     switch (channel)
58     {
59     case CH_RED:
60         for(int i = 0; i < 100; i++)
61         {
62             rs += analogRead(COPIN);
63             counter++;
64             delay(2);
65         }
66         return rs/counter;
67     }
68     return 0;
69 }
70
71 void calibrateMICS()
72 {
73     uint8_t seconds = 10;
74     // Allowed delta for the average from the current value
75     uint8_t delta = 2;
76
77     // Circular buffer for the measurements
78     uint16_t bufferNH3[seconds];
79     uint16_t bufferRED[seconds];
80     uint16_t bufferOX[seconds];
81     // Pointers for the next element in the buffer
82     uint8_t pntrNH3 = 0;
83     uint8_t pntrRED = 0;
84     uint8_t pntrOX = 0;
85     // Current floating sum in the buffer
86     uint16_t fltSumNH3 = 0;
87     uint16_t fltSumRED = 0;
88     uint16_t fltSumOX = 0;
89

```

```

90 // Current measurements;
91 uint16_t curNH3;

92 uint16_t curRED;
93 uint16_t curOX;
94
95 // Flag to see if the channels are stable
96 bool NH3stable = false;
97 bool REDstable = false;
98 bool OXstable = false;
99
100 // Initialize buffer
101 for (int i = 0; i < seconds; ++i)
102 {
103     bufferNH3[i] = 0;
104     bufferRED[i] = 0;
105     bufferOX[i] = 0;
106 }
107
108 do
109 {
110     // Wait a second
111     delay(1000);
112     Serial.print(".");
113     // Read new resistances
114     unsigned long rs = 0;
115
116     delay(50);
117     for (int i = 0; i < 3; i++)
118     {
119         delay(1);
120         rs += analogRead(COPIN);
121     }
122     curRED = rs/3;
123
124
125     fltSumRED = fltSumRED + curRED - bufferRED[pntrRED];
126     bufferRED[pntrRED] = curRED;
127     REDstable = abs(fltSumRED / seconds - curRED) < delta;
128     pntrRED = (pntrRED + 1) % seconds;
129
130     if(!REDstable)
131     {
132         Serial.print("(RED:");
133         Serial.print(abs(fltSumNH3 / seconds - curRED));
134         Serial.print(")");
135     }
136

```

```

137 }
138 while(!REDstable);//while (!NH3stable || !REDstable || !OXstable);
139
140 REDbaseR = fltSumRED / seconds;
141
142 }
143
144 uint16_t getBaseResistance(channel_t channel)
145 {
146     switch (channel)
147     {
148         case CH_RED:
149             return REDbaseR;
150     }
151     return 0;
152 }
153
154 float getCurrentRatio(channel_t channel)
155 {
156     float baseResistance = (float) getBaseResistance(channel);
157     float resistance = (float) getResistance(channel);
158     return resistance / baseResistance * (1023.0 - baseResistance) / (1023.0 -
        resistance);
159     return -1.0;
160 }
161
162 float measureMICS(gas_t gas)
163 {
164     float ratio;
165     float c = 0;
166
167     switch(gas)
168     {
169         case CO:
170             ratio = getCurrentRatio(CH_RED);
171             c = pow(ratio, -1.179) * 4.385;//////// AKI SE CALIBRA EL SENSOR DE
                CO////////
172             break;
173
174         case NO2:
175             ratio = getCurrentRatio(CH_OX);
176             c = pow(ratio, 1.007) / 6.855;
177             break;
178         case NH3:
179             ratio = getCurrentRatio(CH_NH3);
180             c = pow(ratio, -1.67) / 1.47;
181             break;

```



```

230 ///////////////////////////////////////////////////FORMALDEIDO////////////////////////////////////
231 //float ForDehi=0.33;
232 //char ForDehiString[10];
233
234 const int adcPin = 35; // Replace with the actual GPIO pin number
235 int adcValue;
236 float voltage;
237 byte FormDehi;
238 float HCHO;
239 ///////////////////////////////////////////////////
240
241 ///////////////////////////////////////////////////
242 #include <DHT.h> //Cargamos la librería DHT
243 DHT dht(13, DHT22, 22);
244 byte Temperatura;
245 byte Humedad;
246 ///////////////////////////////////////////////////
247
248 ///////////////////////////////////////////////////
249 #define PinRelay 36
250 #define LED 2
251
252 #define IntenDeConec 10
253 byte VarIntenDeConec=0;
254 ///////////////////////////////////////////////////
255 String StringGeneral;
256
257 #define MinutosSensado 5
258 #define MinutosVentilacion 3
259 byte VarMinActual=0;
260 byte VarSegActual=0;
261 bool Ventilador=false;
262 void setup()
263 {
264 ///////////////////////////////////////////////////
265 pinMode(PinRelay,OUTPUT);
266 digitalWrite(PinRelay,HIGH);
267 pinMode(LED,OUTPUT);
268 digitalWrite(LED,HIGH);
269 ///////////////////////////////////////////////////
270
271 ///////////////////////////////////////////////////
272 // put your setup code here, to run once:
273 Serial.begin(9600);
274 delay(10);
275 Serial2.begin(9600);
276 delay(10);
277 Serial.println();

```

```

278 ///////////////////////////////////////////////////////////////////
279
280 ///////////////////////////////////////////////////////////////////
281 Wire.begin();
282 // Enable CCS811
283 ccs811.set_i2cdelay(50); //
284 ok = ccs811.begin();
285 if(!ok)
286 Serial.println("setup: CCS811 begin FAILED");
287 // Start measuring
288 ok = ccs811.start(CCS811_MODE_1SEC);
289 if(!ok)
290 Serial.println("setup: CCS811 start FAILED");
291 ///////////////////////////////////////////////////////////////////
292
293 ///////////////////////////////////////////////////////////////////
294 dht.begin();
295 delay(10);
296 ///////////////////////////////////////////////////////////////////
297
298 ///////////////////////////////////////////////////////////////////
299 Serial.println("MICS-6814 Sensor Test v0.1");
300 Serial.print("Calibrating Sensor");
301 calibrateMICS();
302 Serial.println("OK!");
303 ///////////////////////////////////////////////////////////////////
304
305 ///////////////////////////////////////////////////////////////////
306 WiFi.mode(WIFI_STA);
307 WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
308 Serial.println("-----Conectando a wifi ----- ");
309 Serial.print("Conectando a: ");
310 Serial.println(WIFI_SSID);
311 while(WiFi.status() != WL_CONNECTED)
312 {
313 Serial.print(".");
314 digitalWrite(LED, HIGH);
315 delay(250);
316 digitalWrite(LED, LOW);
317 delay(250);
318 VarIntenDeConec++;
319 if(VarIntenDeConec>=IntenDeConec)
320 break;
321
322 }
323 // Verificar si la conexión fue exitosa
324 if (WiFi.status() != WL_CONNECTED)
325 {

```

```

326 Serial.println("No se pudo conectar al WiFi.");
327 }
328 else
329 {
330 Serial.println(" Conectado a WiFi. Dirección IP: ");
331 Serial.println(WIFI_SSID);
332 Serial.print("IP asignada: ");
333 Serial.println(WiFi.localIP()); // Imprimir la dirección IP obtenida
334 Serial.println("----- ");
335 //}
336
337 // Asignar la API Key (requerido)
338 config.api_key = API_KEY;
339 // Asignar la URL de la base de datos RTDB (requerido)
340 config.database_url = DATABASE_URL;
341 // Registrarse en Firebase
342 Serial.println();
343 Serial.println("--Registrandose-----");
344 Serial.print("Nuevo usuario... ");
345 if (Firebase.signUp(&config, &auth, "", ""))
346 {
347 Serial.println("ok");
348 signupOK = true;
349 }
350 else
351 {
352 Serial.printf("%s\n", config.signer.signupError.message.c_str());
353 }
354 Serial.println("-----");
355 // Asignar la funcion de callback para el estado del token
356 config.token_status_callback = tokenStatusCallback;
357 Firebase.begin(&config, &auth);
358 Firebase.reconnectWiFi(true);
359 }
360 ///////////////////////////////////////////////////////////////////
361 }
362
363 void loop()
364 {
365 delay(5000);
366 ///////////////////////////////////////////////////////////////////
367 while(1)
368 {
369 if(pms.read(data))
370 {
371 //Serial.println("Dust Concentration");
372 PM25 = data.PM_AE_UG_2_5;
373 //NUEVOPM25 = map(PM25, 5, 50, 0, 25);

```

```

374     Serial.print("PM2.5 :");Serial.print(PM25,1);Serial.println("(ug/m3)");
375
376     break;
377 }
378 }
379 ////////////////////////////////////////////////////
380 Co=measureMICS(CO);
381 Serial.print("CO:");
382 Serial.print(Co);
383 Serial.println("ppm");
384 ////////////////////////////////////////////////////
385 uint16_t eco2, etvoc, errstat, raw;
386 ccs811.read(&eco2, &etvoc, &errstat, &raw);
387 // Print measurement results based on status
388 if (errstat == CCS811_ERRSTAT_OK)
389 {
390     Co2 = eco2;
391
392     Serial.print("CCS811: ");
393     Serial.print("Co2=");
394     Serial.print(Co2);
395     Serial.println(" ppm ");
396 }
397 else if(errstat == CCS811_ERRSTAT_OK_NODATA)
398 {
399     Serial.println("CCS811: waiting for (new) data");
400 }
401 else if(errstat & CCS811_ERRSTAT_I2CFAIL)
402 {
403     Serial.println("CCS811: I2C error");
404 }
405 else
406 {
407     Serial.print("CCS811: errstat=");
408     Serial.print(errstat, HEX);
409     Serial.print("=");
410     Serial.println( ccs811.errstat_str(errstat) );
411 }
412 ////////////////////////////////////////////////////
413 adcValue = analogRead(adcPin);
414
415 voltage = (float)adcValue / 4096.0 * 3.3; //TAKI LA CALIBRACION DEL
FORMALDEIDO//
416
417 if(voltage >= 3.22)
418     FormDehi = 10;
419 else if(voltage >= 3.16)
420     FormDehi = 9;

```



```

468 Serial.println();
469 if(Ventilador == false)
470 {
471     if(PM25 >= 15 || Co >= 20 || FormDehi >= 0.3 || Co2 >= 600)
472     {
473         VarSegActual=VarSegActual+5;
474         if(VarSegActual <= 60)
475         {
476             VarMinActual=VarMinActual+1;
477             VarSegActual=0;
478             if(VarMinActual >= MinutosSensado)
479             {
480                 digitalWrite(PinRelay,LOW);
481                 Ventilador = true;
482                 VarSegActual=0;
483                 VarMinActual=0;
484             }
485         }
486     }
487     else
488     {
489         VarSegActual=0;
490         VarMinActual=0;
491     }
492 }
493 else
494 {
495     VarSegActual=VarSegActual+5;
496     if(VarSegActual <= 60)
497     {
498         VarMinActual=VarMinActual+1;
499         VarSegActual=0;
500         if(VarMinActual >= MinutosVentilacion)
501         {
502             digitalWrite(PinRelay,HIGH);
503             Ventilador = false;
504             VarSegActual=0;
505             VarMinActual=0;
506         }
507     }
508 }
509 }
510
511 void EnviaDatoWeb()/////aumentar aki en caso de sumar
tachos////////////////////////////////////
512 {
513     Serial.print("Enviando Dato de Sensores ");
514     //digitalWrite(LED, HIGH);

```

```

515
516 if (Firebase.ready() && signupOK )
517 {
518     digitalWrite(LED, HIGH);
519     ///////////////DUPLICAR ESTO PERO CAMBIANDO EL NOMBRE Y EL
ESTADO/////////////////
520     dtostrf(PM25,4,1,PM25String);//convierte los datos float del PM25 a una
cadena de caracteres y lo guarda en PM25String con ancho de 4 y 1
decimal
521     StringGeneral = PM25String;
522     StringGeneral += " ug/m3";
569     Serial.println("PASSED");
570     Serial.println("PATH: " + fbdo.dataPath());
571     Serial.println("TYPE: " + fbdo.dataType());
572 }
573 else
574 {
575     Serial.println("FAILED");
576     Serial.println("REASON: " + fbdo.errorReason());
577 }
578     //////////////////////////////////////
579     StringGeneral = String(Temperatura);
580     StringGeneral += " gC";
581     if(Firebase.RTDB.setString(&fbdo,"Temp",StringGeneral))//if(Firebase.RTD
B.setInt(&fbdo,"Temp",Temperatura))
582     {
583         Serial.println("PASSED");
584         Serial.println("PATH: " + fbdo.dataPath());
585         Serial.println("TYPE: " + fbdo.dataType());
586     }
587     else
588     {
589         Serial.println("FAILED");
590         Serial.println("REASON: " + fbdo.errorReason());
591     }
592     //////////////////////////////////////
593     StringGeneral = String(Humedad);
594     StringGeneral += " %";
595     if(Firebase.RTDB.setString(&fbdo,"HR",StringGeneral))// if(Firebase.R
TDB.setInt(&fbdo,"HR",Humedad))
596     {
597         Serial.println("PASSED");
598         Serial.println("PATH: " + fbdo.dataPath());
599         Serial.println("TYPE: " + fbdo.dataType());
600     }
601     else
602     {
603         Serial.println("FAILED");
604         Serial.println("REASON: " + fbdo.errorReason());

```

```
605 }
606 delay(500);
607 digitalWrite(LED, LOW);
608 }
609 else
610   Serial.println("Fire Base No: Firebase.ready() && signupOK");
611 //digitalWrite(LED, LOW);
612 }
```