

**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE INGENIERÍA**

**CARRERA DE COMPUTACIÓN**

**TEMA:**

**Interfaz gráfica de monitoreo de la calidad del aire en un sector  
residencial de Guayaquil**

**AUTOR:**

**Burbano Delgado María Paula**

**Trabajo de integración curricular previo a la obtención del título de  
INGENIERO EN CIENCIAS DE LA COMPUTACIÓN**

**TUTOR:**

**Ing. Célleri Mujica, Colon Mario**

**Guayaquil, Ecuador**

**02 de marzo del 2022**



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE INGENIERÍA**

**CARRERA DE COMPUTACIÓN**

## **CERTIFICACIÓN**

Certificamos que el presente trabajo de integración curricular fue realizado en su totalidad por **Burbano Delgado María Paula**, como requerimiento para la obtención del título de **Ingeniero en Ciencias de la Computación**.

**TUTOR**

f.

\_\_\_\_\_  
Ing. Céleri Mujica, Mario Colón, Mgs.

**Guayaquil, a los 02 días del mes de marzo del año 2022**



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE INGENIERÍA**

**CARRERA DE COMPUTACIÓN**

## **DECLARACIÓN DE RESPONSABILIDAD**

Yo, **Burbano Delgado María Paula**

### **DECLARO QUE:**

El Trabajo de Integración Curricular, **Interfaz gráfica de monitoreo de la calidad del aire en un sector residencial de Guayaquil**, previo a la obtención del título de Ingeniero en Ciencias de la Computación, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Integración Curricular referido.

**Guayaquil, a los 02 días del mes de marzo del año 2022**

f. 

Nombre: **Burbano Delgado, María Paula**



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE INGENIERÍA**

**CARRERA DE COMPUTACIÓN**

**AUTORIZACIÓN**

Yo, **Burbano Delgado María Paula**

Autorizo a la Universidad Católica de Santiago de Guayaquil a la publicación en la biblioteca de la institución del Trabajo de Integración Curricular, **Interfaz gráfica de monitoreo de la calidad del aire en un sector residencial de Guayaquil**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

**Guayaquil, a los 02 días del mes de marzo del año 2022**

**EL AUTOR:**

f. 

Nombre: **Burbano Delgado, María Paula**



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE INGENIERIA

CARRERA DE COMPUTACIÓN

REPORTE URKUND

← → ↻ secure.orkund.com/old/view/121698854-202969-744979#q1bKLVayjibQMdQx0ji

URKUND

Documento	<a href="#">Titulación Burbano.pdf</a> (D127565245)
Presentado	2022-02-10 09:45 (-05:00)
Presentado por	maria.burbano04@cu.ucsg.edu.ec
Recibido	colon.celleri.ucsg@analysis.orkund.com
Mensaje	TESIS <a href="#">Mostrar el mensaje completo</a>

2% de estas 21 páginas, se componen de texto presente en 6 fuentes.

TUTOR

f. \_\_\_\_\_

Ing. Céleri Mujica, Mario Colón, Mgs.

## **AGRADECIMIENTO**

Me gustaría agradecer por el presente trabajo a mis padres, Juan Carlos y Alina, por guiarme en la vida y darme todas las oportunidades que he tenido.

Estoy agradecida también por toparme con personas tan maravillosas como mis compañeros, cada uno de ellos ha formado parte importante de este camino.

Quisiera hacer una mención especial a Katherine y Rebeca, gracias por ser mis amigas desde el primer día y ser mi grupo de apoyo.

Gracias a mi tutor el Ing. Celleri por guiarme en la realización de este trabajo, así como al resto de mis profesores que me compartieron su conocimiento a lo largo de estos años.

Finalmente quisiera agradecerle a Leandro, el amor de mi vida, por su apoyo incondicional y por convertir los días difíciles en felicidad.

Mi gratitud infinita con todos los involucrados en este proceso, me llevo cada enseñanza y recuerdo conmigo.

**Burbano Delgado María Paula**

## **DEDICATORIA**

A mi papa, por su amor y por siempre estar cuando lo necesito.

A mi mama, por su cariño y sus consejos que me ayudaron a cumplir mis metas.

A mi tío Lebe, que me inspiro su fortaleza y siempre estará en mi corazón.

A mí misma, por no rendirme ante las adversidades y ser capaz de lograr mi  
objetivo.

**Burbano Delgado María Paula**

# ÍNDICE GENERAL

DEDICATORIA.....	VII
RESUMEN .....	XV
ABSTRACT .....	XVI
Introducción.....	2
CAPÍTULO I .....	5
EL PROBLEMA.....	5
1.1. Planteamiento del problema.....	5
1.2. Delimitación.....	6
1.3. Pregunta de investigación.....	7
1.4. Objetivos de la investigación.....	8
1.4.1. Objetivo General.....	8
1.4.2. Objetivos específicos.....	8
1.5. Justificación y Alcance.....	8
1.5.1. Justificación .....	8
1.5.2. Alcance.....	9
CAPITULO II .....	11
MARCO TEÓRICO Y REFERENCIAL .....	11
2.1. Contaminación ambiental.....	11
2.2. Efectos en la salud.....	12
2.3. Tipos de sensores para monitoreo del CO2.....	12
2.3.1. Sensor MQ135 .....	14
2.4. Interfaz gráfica .....	15
2.4.1. Power BI.....	16
2.5. Elementos del hardware .....	20

2.5.1. Arduino Nano.....	20
2.5.2. Cargadores de 5v 2A.....	22
2.5.3. Convertidor TTL.....	22
2.5.4. Esp32 .....	23
2.5.5. Placa de circuito PCB.....	24
2.5.6. Switch .....	25
2.5.7. Cables jumper .....	26
2.6. Comunicación entre hardware, software y Power BI.....	27
2.7. Casos de uso .....	32
2.7.1. Monitoreo del aire para prevenir contagios de COVID-19.....	32
2.7.2. Monitoreo del aire para exteriores .....	34
CAPITULO III .....	36
METODOLOGÍA Y RESULTADOS.....	36
3.1. Metodología de la investigación.....	36
3.1.1. Recolección de datos.....	38
3.2. Metodología de desarrollo .....	39
3.2.1. Metodología de desarrollo por prototipos.....	39
3.2.2. Metodología de sistema embebido .....	50
3.3. Análisis de resultados.....	51
3.3.1. Análisis de las entrevistas.....	52
3.3.2. Análisis de la calidad del aire por medio de solución IoT implementada	55
3.3.3. Análisis de la situación actual del monitoreo del aire en Guayaquil .....	57
CAPITULO IV.....	59
PROPUESTA.....	59
4.1. Objetivo .....	59

4.2. Responsable.....	59
4.3. Descripción de la solución IoT.....	60
4.4. Herramientas tecnológicas.....	60
4.5. Proceso.....	61
4.6. Requerimientos de implementación.....	69
4.6.1. Hardware.....	69
4.6.2. Software.....	70
4.6.3. Servidor.....	70
4.7. Materiales.....	71
4.8. Estudio Costo/Beneficio.....	72
4.9. Resultados esperados.....	73
CONCLUSIONES.....	74
RECOMENDACIONES.....	75
REFERENCIAS BIBLIOGRÁFICAS.....	76
APENDICES.....	80
Apéndice A. Formato de ENTREVISTA: El CO2 y sus efectos nocivos en la salud y el ambiente.....	80
Apéndice B. Formato de ENTREVISTA: El internet de las cosas y la contaminación ambiental.....	81
ANEXOS.....	82

## Índice de figuras

Figura 1. Mapa geográfico de la ciudadela Ceibos Norte. Nota Tomada de: Administración Ceibos Norte, 2021 .....	7
Figura 2. Sensor infrarrojo MH-Z19B de CO2. Nota Tomado de: Unit Electronics, s.f. ....	13
Figura 3. Sensor electroquímico MQ135. Nota Tomado de: MarkerElectronico, 2022 .....	13
Figura 4. Sensor semiconductore de oxido metálico. Nota Tomado de: DirectIndustry .....	14
Figura 5. Sensor MQ135. Nota Tomado de: MarkerElectronico.....	15
Figura 6. Power BI Desktop .....	18
Figura 7. Servicio Power BI.....	18
Figura 8. Aplicación móvil de Power BI .....	19
Figura 9. Conversión Analógica – Digital. Nota Tomado de: Control Automático Educación .....	21
Figura 10. Cargador de 5V 2A. Nota Tomado de: Novatronic.....	22
Figura 11. Convertidor TTL. Nota Tomado de: Aliexpress .....	23
Figura 12. Placa ESP32. Nota Tomado de: CDMX Electrónica .....	24
Figura 13. Placa de circuito PCB. Nota Tomado de: Instructables.....	25
Figura 14. Interruptor basculante. Nota Tomado de: Codrey Electronics.....	26
Figura 15. Diferentes tipos de cables jumper. Nota Tomado de: Electrónica Caribe	27
Figura 16. Partes del sensor MQ135. Nota Tomado de: Iberotecno .....	28
Figura 17. Partes de la placa Arduino Nano. Nota Tomado de: Roboteach.....	29
Figura 18. Comando readSensor dentro del código.....	29
Figura 19. Conexiones del hardware de los prototipos. ....	30
Figura 20. Base de datos disponible en Power BI.....	32
Figura 21. Campos utilizados para visualización del mapa.....	32

Figura 22. Variación de CO2 en el aire de las aulas. Nota Tomado de: Elsevier .....	34
Figura 23. Monitoreo de CO2 en diferentes partes del mundo. Nota Tomado de: uRADMonitor.....	35
Figura 24. Arquitectura de la comunicación IoT hasta Power BI .....	41
Figura 25. Arquitectura del hardware .....	42
Figura 26. Parte superior del prototipo IoT .....	43
Figura 27. Parte inferior del prototipo IoT .....	43
Figura 28. Exterior del prototipo. ....	44
Figura 29. Parte del código para la placa Arduino Nano .....	44
Figura 30. Parte del código para la placa ESP32.....	45
Figura 31. Datos en el monitor serie del IDE Arduino .....	46
Figura 32. Prototipo conectado a enchufe eléctrico .....	47
Figura 33. Ingreso al portal de servicios de Aurora.....	48
Figura 34. Prototipos A1, A2, A3 en el portal de servicios de Aurora.....	48
Figura 35. Detalle de eventos en Aurora.....	49
Figura 36. Historial del sensor AP3.....	49
Figura 37. Mapa de Ceibos Norte con los puntos de los prototipos señalados.....	50
Figura 38. Gráficos del CO2 a través del tiempo .....	50
Figura 39. Metodología de sistema embebido .....	51
Figura 40. Entrevista al Ing. Diego Eraso vía e-mail. ....	53
Figura 41. Entrevista con el Ing. Giancarlo Molina por videollamada.....	55
Figura 42. Resultados del prototipo AP1 .....	56
Figura 43. Resultados del prototipo AP2.....	57
Figura 44. Resultados del prototipo AP3.....	57
Figura 45. Punto de ubicación del sensor AP1. ....	62
Figura 46. Punto de ubicación del sensor AP2. ....	62

Figura 47. Punto de ubicación del sensor AP3. ....	63
Figura 48. AP1 instalado .....	64
Figura 49. AP2 instalado .....	64
Figura 50. AP3 instalado .....	65
Figura 51. Software interpretador de Aurora .....	66
Figura 52. Base de datos SQL Server de Aurora .....	66
Figura 53. Comunicación de la solución IoT .....	67
Figura 54. Pantalla de Power BI Desktop.....	68
Figura 55. Pantalla del servicio de Power BI Pro .....	69
Figura 56. Parte 1 del código del programa de Arduino Nano .....	82
Figura 57. Parte 2 del código del programa de Arduino Nano .....	82
Figura 58. Parte 3 del código del programa de Arduino Nano .....	83
Figura 59. Parte 4 del código del programa de Arduino Nano .....	83
Figura 60. Parte 5 del código del programa de Arduino Nano .....	84
Figura 61. Parte 1 del código del programa de ESP32 .....	84
Figura 62. Parte 2 del código del programa de ESP32 .....	84
Figura 63. Parte 3 del código del programa de ESP32 .....	85
Figura 64. Parte 4 del código del programa de ESP32 .....	85
Figura 65. Parte 5 del código del programa de ESP32 .....	85
Figura 66. Parte 6 del código del programa de ESP32 .....	86
Figura 67. Parte 7 del código del programa de ESP32 .....	86
Figura 68. Parte 8 del código del programa de ESP32 .....	86

## Índice de tablas

Tabla 1. Materiales para la elaboración del hardware.....	71
Tabla 2. Materiales para la implementación del software.....	71
Tabla 3. Costo total de materiales para implementación de solución IoT .....	72

## RESUMEN

El papel que desarrollan los humanos en la tierra es la principal razón de que la contaminación cada vez sea más preocupante. Dado que esta situación no ha mejorado en las últimas décadas, se ha vuelto imprescindible el control y monitoreo de la calidad del aire en el ambiente. Los monitoreos del aire se realizan para asegurar la salud de los ciudadanos al igual que la conservación de la flora y fauna que nos rodea, también es una manera de evitar el deterioro urbanístico de las ciudades. El objetivo de este proyecto es monitorear el aire de la ciudadela Ceibos Norte, mediante una solución IoT utilizando las placas Arduino Nano y ESP32. El servidor que se escogió para enviar la información recolectada es Aurora, y la herramienta para la visualización de los resultados de manera grafica fue Power BI. Para la realización del actual proyecto se utilizó una metodología mixta, dándole uso a entrevistas y sacando conclusiones de los valores que los prototipos IoT arrojan. Mediante las entrevistas se pudo concluir que se necesita concientizar más a la población ecuatoriana sobre la importancia de la conservación ambiental y como los avances tecnológicos pueden ayudarnos a reducir el impacto negativo que tenemos sobre el medio ambiente. Además, se utilizó la metodología de diseño rápido para el prototipado donde se definió la arquitectura del mismo, lo que fue de ayuda para realizar los diferentes prototipos de manera correcta. El conjunto de las metodologías dio resultados positivos y satisfactorios para la solución IoT del presente trabajo de titulación.

**Palabras clave:** *Arduino Nano, placa ESP32, sensor MQ135, IOT, Aurora, Power BI.*

## ABSTRACT

The role that humans play on earth is the main reason why pollution is becoming more and more of a concern. Since this situation has not improved in recent decades, the control and monitoring of air quality in the environment has become essential. Air monitoring is carried out to ensure the health of citizens as well as the conservation of the flora and fauna that surrounds us, it is also a way to avoid the urban deterioration of cities. The objective of this project is to monitor the air of the Ceibos Norte citadel, through an IoT solution using the Arduino Nano and ESP32 boards. The server that was chosen to send the collected information is Aurora, and the tool for visualizing the results graphically was Power BI. To carry out the current project, a mixed methodology was obtained, interviews were offered and conclusions were drawn from the values that the IoT prototypes yielded. Through the interviews it was possible to conclude that the Ecuadorian population needs to become more aware of the importance of environmental conservation and how technological advances can help us reduce the negative impact we have on the environment. In addition, the rapid design methodology for prototyping was produced, where its architecture was defined, which was helpful in making the different prototypes correctly. The set of methodologies gave positive and satisfactory results for the IoT solution of this degree work.

**Keywords:** *Arduino Nano, ESP32 board, MQ135 sensor, IOT, Aurora, Power BI.*

## Introducción

Al hablar de contaminación ambiental nos referimos a “el ingreso de sustancias químicas nocivas en un entorno determinado. Este fenómeno repercute en el equilibrio de dicho entorno y lo convierte en un ambiente inseguro” (Fundación Aquae, s.f.). El papel que desarrollan los humanos en la tierra es la principal razón de que la contaminación cada vez sea más preocupante. Dado que esta situación no ha mejorado en las últimas décadas, se ha vuelto imprescindible el control y monitoreo de la calidad del aire en el ambiente. Los monitoreos del aire se realizan para asegurar la salud de los ciudadanos al igual que la conservación de la flora y fauna que nos rodea, también es una manera de evitar el deterioro urbanístico de las ciudades.

Otro tema importante que es consecuencia de los gases nocivos en el medio ambiente es el cambio climático. El cambio climático, también conocido como calentamiento global, se da por los gases que atrapan el calor en la atmósfera, estos gases se denominan gases de efecto invernadero. Los principales gases de efecto invernadero son el dióxido de carbono, metano, óxido nitroso y gases fluorados (Pachauri & Reisinger, 2008).

El dióxido de carbono (CO<sub>2</sub>) es el que se encuentra en mayor abundancia. Ingresa a la atmósfera a través de la quema de combustibles fósiles (carbón, gas natural y petróleo), desechos sólidos, árboles y otros materiales biológicos, y también como resultado de ciertas reacciones químicas como la fabricación de cemento. El dióxido de carbono se elimina de la atmósfera cuando es absorbido por las plantas como parte del ciclo biológico del carbono.

El segundo más abundante es el metano (CH<sub>4</sub>), el cual se emite durante la producción y el transporte de carbón, gas natural y petróleo. Las emisiones de metano también son el resultado de la ganadería, el uso de la tierra y la descomposición de los desechos orgánicos en los vertederos de desechos sólidos municipales.

También está presente el óxido nitroso (N<sub>2</sub>O), que se emite durante actividades agrícolas, de uso de la tierra, industriales, combustión de combustibles fósiles y residuos sólidos, así como durante el tratamiento de aguas residuales.

Finalmente, y en menor cantidad están los gases fluorados, que es un grupo de diferentes gases, entre estos: los hidrofluorocarbonos, los perfluorocarbonos, el hexafluoruro de azufre y el trifluoruro de nitrógeno. Todos hacen parte de los gases de efecto invernadero sintéticos que se emiten a partir de una variedad de procesos industriales. Los gases fluorados a veces se utilizan como sustitutos de sustancias que agotan el ozono estratosférico. Por lo general, estos gases se emiten en cantidades más pequeñas, pero debido a que son potentes gases de efecto invernadero, a veces se los denomina gases de alto potencial de calentamiento global (Pachauri & Reisinger, 2008).

Según la Agencia de Protección Ambiental (EPA), Un vehículo de pasajeros típico emite aproximadamente 4,6 toneladas métricas de dióxido de carbono por año. Este número puede variar según el combustible de un vehículo, la economía de combustible y la cantidad de millas recorridas por año.

Conducir la mayoría de los vehículos genera emisiones de gases de efecto invernadero en el tubo de escape. La producción y distribución del combustible que

se utiliza para impulsar un vehículo también genera gases de efecto invernadero. La gasolina, por ejemplo, requiere extraer petróleo del suelo, transportarlo a una refinería, refinar el petróleo para convertirlo en gasolina y transportar la gasolina a las estaciones de servicio. Cada uno de estos pasos puede producir emisiones adicionales de gases de efecto invernadero (EPA, s.f.).

# **CAPÍTULO I**

## **EL PROBLEMA**

En este capítulo se expondrá la problemática identificada en la ciudadela residencial Ceibos Norte, donde existe la necesidad de monitorear la cantidad de CO<sub>2</sub> en el aire para revelar si esta está entre los límites seguros.

### **1.1. Planteamiento del problema.**

Actualmente Ecuador cuenta con 17,64 millones de habitantes, de los cuales se estima que emiten 1,9 toneladas métricas de CO<sub>2</sub> cada uno. Ya que nuestro país no cuenta con un gran número de habitantes comparado a otros a nivel mundial, representamos menos del 0,5% de emisiones a nivel mundial.

A pesar de que no representamos un peligro mundialmente si necesitamos reducir la contaminación a nivel nacional, ya que entre los años 1990 y 2006 se ha reportado un aumento del 78.7% en las emisiones de dióxido de carbono en consecuencia de los transportes (MAE, s.f.).

En el presente trabajo se monitoreará únicamente el dióxido de carbono en el sector residencial Ceibos Norte. El tema central estará enfocado en una ciudadela cerrada, por lo que el contaminante que más destacara serán los carros de los propietarios y sus visitantes. A pesar de que Ceibos Norte no es una ciudadela grande, existe una gran cantidad de carros en la zona, estacionados o andando que claramente podrían afectar la calidad del aire que respiran los que la habitan.

Para el análisis de la información resultante de este trabajo se hace referencia a el Departamento de Servicios de Salud de Wisconsin, donde se establecen los niveles normales o tolerables del CO<sub>2</sub>. Para el exterior, un nivel normal oscila entre 300 a 400 ppm (partes por millón) y para lugares cerrados como las casas se suelen presentar concentración de 400 a 800 ppm. Cantidades mayores a estas suelen estar asociadas con dolores de cabeza, somnolencia, aumento de la frecuencia cardíaca y náuseas leves (Wisconsin Department of Health Services, 2021).

## **1.2. Delimitación**

Para desarrollar este proyecto se ha decidido delimitar el monitoreo a un área específica de la ciudad de Guayaquil, la ciudadela Ceibos Norte que se encuentra en el Km. 30.5 Vía Perimetral. Este conjunto residencial contaba con 900 casas, 1000 familias y 4800 habitantes hasta el año 2017 (Asociacion de Propietarios de la Urbanizacion Ceibos Norte, 2017).

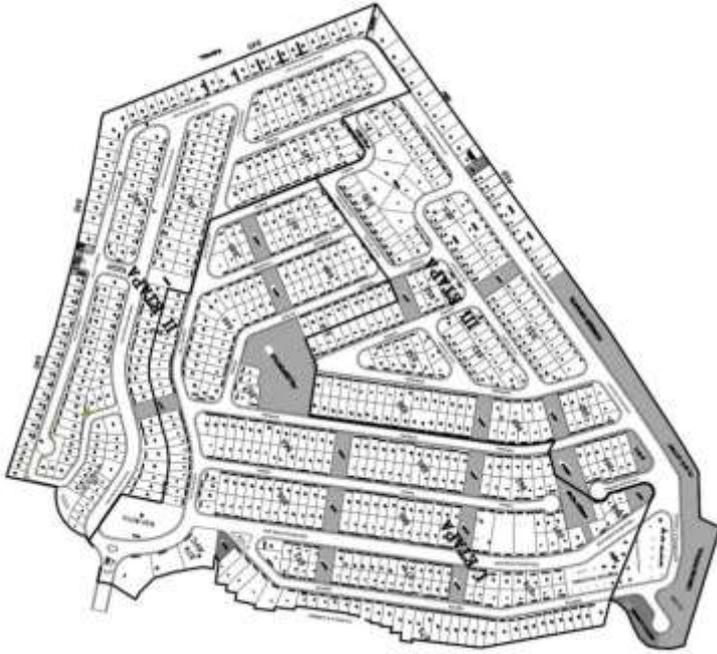


Figura 1. Mapa geográfico de la ciudadela Ceibos Norte. Nota Tomada de: Administración Ceibos Norte, 2021

La urbanización cuenta con dos garitas. La primera se encuentra dentro de otra ciudadela llamada Cumbres. La segunda garita se encuentra en la Vía Perimetral por donde se llegan a concentrar grandes cantidades de carros, motos, buses y camiones.

### **1.3. Pregunta de investigación**

La pregunta planteada para este trabajo de investigación es la siguiente:

¿Los niveles de dióxido de carbono en Ceibos Norte están dentro de los límites normales?

## **1.4. Objetivos de la investigación**

Los objetivos que se plantean para solucionar la problemática expuesta anteriormente son:

### **1.4.1. Objetivo General**

Desarrollar una interfaz gráfica que monitore la calidad del aire en un sector residencial de Guayaquil.

### **1.4.2. Objetivos específicos**

- Establecer los puntos en los que se encontraran los sensores dentro de la ciudadela.
- Diseñar e implementar el hardware del sistema con las placas Arduino y ESP32.
- Monitorear la calidad del aire en tiempo real.
- Mostrar los datos en un mapa donde se podrán visualizar los tres puntos y sus respectivos valores de CO<sub>2</sub>.

## **1.5. Justificación y Alcance**

En esta sección se presentará la justificación y el alcance del presente trabajo de integración.

### **1.5.1. Justificación**

Como ya se había señalado antes, Ceibos Norte cuenta con 900 casas en las cuales la mayoría cuenta con una o dos casas. Las calles de esta ciudadela siempre

están llenas de carros y hay calles donde no hay mucho espacio para parquearse. Aparte de esto, los residentes suelen optar por usar sus vehículos en vez de caminar. A pesar de que es un conjunto residencial pequeño y las tiendas que existen dentro no se encuentran a más de 5 minutos caminando.

Existe otro punto que podría ser muy contaminante de este sector, y es la Vía Perimetral por la que pasan grandes cantidades de vehículos de todo tipo todos los días a toda hora. Esta se encuentra justo al lado de la segunda garita de la ciudadela.

Por estas razones creo importante el monitoreo del aire enfocado en el dióxido de carbono, dado los efectos que puede tener para la salud de los residentes y para todos los animales que habitan esta zona.

### **1.5.2. Alcance**

El monitoreo de la calidad del aire en la ciudadela Ceibos Norte tendrá como alcance los siguientes puntos:

- Los datos serán mostrados a través de Power BI Pro, actualizándose cada hora.
- El sensor y el programa tendrán la tarea de medir únicamente el dióxido de carbono en el aire.
- El sensor va a medir y enviar datos al servidor cada un minuto, la medición será continua, pero con pausas.

- La finalidad de este proyecto no es llegar a monitorear el aire a nivel nacional, si no ser una opción más económica para monitorear la cantidad de CO<sub>2</sub> en pequeños sectores como las ciudadelas.

## **CAPITULO II**

### **MARCO TEÓRICO Y REFERENCIAL**

#### **2.1. Contaminación ambiental**

La contaminación del aire se origina en numerosas fuentes de emisión, tanto naturales como derivadas de la actividad del hombre, con este último convirtiéndose en globalmente dominante desde el comienzo de la industrialización. El proceso de la combustión es el mayor contribuyente a la contaminación del aire, en particular, la combustión de combustibles fósiles y biomasa para generar energía. Las fuentes de combustión al aire libre incluyen los medios de transporte como carros, aviones y botes; generación de energía; y quema de biomasa, que incluye los incendios controlados y los no controlados de bosques, así como quema de desechos en áreas urbanas (World Health Organization, 2021).

Uno de los gases más contaminantes es el dióxido de carbono (CO<sub>2</sub>). Se encuentra en la naturaleza, pero por consecuencia de la realización continua de las actividades realizadas anteriormente este gas aumenta considerablemente llegando a niveles que llegan a ser nocivos para los seres vivos y el medio ambiente. En mayo del 2021 los niveles de CO<sub>2</sub> alcanzaron su punto máximo con un promedio mensual de 419 partes por millón (ppm), el nivel más alto desde que comenzaron las mediciones precisas hace 63 años ( Mauna Loa Atmospheric Baseline Observatory, 2021).

## **2.2. Efectos en la salud**

Casi toda la población mundial (99%) está expuesta a niveles de contaminación del aire que los ponen en mayor riesgo de enfermedades que incluyen enfermedades cardíacas, accidentes cerebrovasculares, enfermedad pulmonar obstructiva crónica, cáncer y neumonía. A continuación, se presentan otros datos alarmantes que tiene sobre la salud:

- Muertes de hasta siete millones de seres humanos al año.
- 2.6 mil millones de personas dependen de tecnología contaminante para cocinar.
- Para cocinar, 2.6 mil millones de nosotros usamos tecnología dañina para el medio ambiente.
- El 91% de las personas a nivel mundial viven en zonas donde el aire se piensa que no es lo suficientemente limpio (World Health Organization, s.f.).

## **2.3. Tipos de sensores para monitoreo del CO2**

Existen 3 tipos de sensores que miden la calidad del aire, estos son: sensores infrarrojos no dispersivos, sensores electroquímicos, sensores semiconductores de oxido metálico.

Los sensores infrarrojos analizan el aire emitiendo luces con ondas específicas. Estas luces son absorbidas por el dióxido de carbono, por lo que la luz absorbida es igual a la cantidad de CO2 presente.

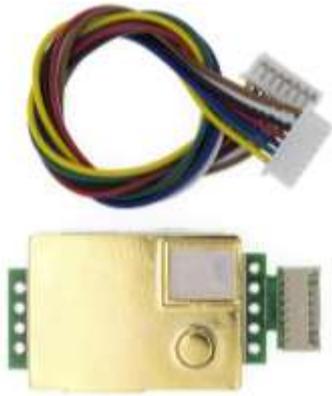


Figura 2. Sensor infrarrojo MH-Z19B de CO2. Nota Tomado de: Unit Electronics, s.f.

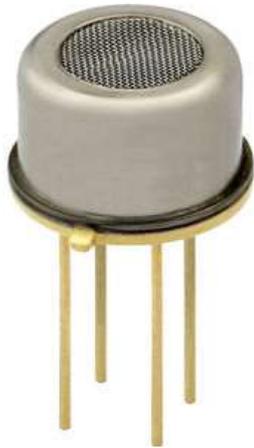
Por otro lado, tenemos los sensores electroquímicos que disponen de un calentador que aumentan su temperatura cuando se ven expuestos diferentes tipos de gases. A consecuencia de esto, el sensor experimenta un cambio eléctrico. Luego, el sensor usará el tipo y la cantidad de cambio eléctrico para determinar la cantidad de CO2 presente.

Este tipo de sensores necesitan tiempo para llegar a una temperatura estable y presentar datos confiables. Se suele recomendar hasta 24 horas en marcha para que la información que nos presente sea 100% confiable.



Figura 3. Sensor electroquímico MQ135. Nota Tomado de: MarkerElectronico, 2022

Finalmente tenemos los semiconductores de óxido metálico, que utilizan la facilidad con la que la electricidad fluye a través de los compuestos metálicos para probar las cantidades de gas en el aire (Smith, 2019).



*Figura 4.* Sensor semiconductore de oxido metálico. *Nota* Tomado de: DirectIndustry

### **2.3.1. Sensor MQ135**

El sensor que se usara en este proyecto es el MQ135, el cual es un sensor electroquímico. Este sensor permite medir varios gases diferentes, entre estos el CO<sub>2</sub>, NH<sub>3</sub>, benceno, entre otros. Pero para este trabajo nos vamos a enfocar únicamente en el CO<sub>2</sub>.



Figura 5. Sensor MQ135. Nota Tomado de: MarkerElectronico

El MQ135 tiene un calefactor interno que tiene la función de mantener una temperatura y humedad estable en el encapsulado que de no ser reguladas afectarían la sensibilidad del sensor. También posee una pequeña malla en el exterior y su trabajo es no permitir que pequeñas partículas entren al dispositivo y lo dañen.

Entre las características de este sensor del dióxido de carbono podemos destacar que es estable y tiene una larga vida útil si se usa de manera adecuada, también es fácil de usar y nos genera respuestas rápidas (iberotecno, 2021).

A pesar de no ser el mejor sensor ni el más preciso para medir este tipo de gases si es uno de los menos costosos. Para este proyecto se llegó a conseguir a \$5,4 cada uno, un precio bastante razonable para la calidad del sensor.

## **2.4. Interfaz gráfica**

La interfaz gráfica de usuario de un programa presenta una pantalla visual fácil de usar para el usuario. Está formado por componentes gráficos como botones, etiquetas, ventanas, etc. a través de estos componentes el usuario puede interactuar con la página o aplicación (Leahy, 2019).

La interfaz gráfica para este proyecto constara de un mapa de la ciudadela Ceibos Norte, en este mapa se podrán apreciar los 3 diferentes sensores. Los sensores se establecerán con las coordenadas de latitud y longitud.

Adicionalmente, se presentarán los datos en diferentes gráficos como tablas, gráficos de anillos, y gráficos de líneas.

La herramienta que se utilizará para poder visualizar los datos será Power BI, que sirve para analizar datos de manera gráfica y les facilita a los usuarios la toma de decisiones (Microsoft, 2021).

#### **2.4.1. Power BI**

La historia de esta herramienta empieza en el año 2010 donde fue nombrada Proyecto Crescent por Ron George y estuvo disponible para ser descargada en el 2011. Unos años más tarde fue cambiaron su nombre por el que tiene en la actualidad y Microsoft lo dio a conocer desde el 2013 como Power BI para Office 365 (The Power BI Team, 2015).

Power BI es una plataforma de Business Intelligence de Microsoft. Convierte fuentes de datos no relacionados en información con sentido y nos permite visualizar esta información de manera interactiva.

Los datos se pueden extraer de diversas fuentes como Excel y bases de datos. Esta herramienta permite extraer la data fácilmente, visualizarla y así poder descubrir información importante (Microsoft, 2021).

#### 2.4.1.1. *Business Intelligence*

Business Intelligence o la inteligencia de negocios ayuda a las empresas a descomponer datos, para que puedan descubrir rápidamente información útil para tomar decisiones. BI procesa extensas cantidades de información y la exhibe de manera gráfica, lo que hace que los usuarios lo entiendan con mayor facilidad.

En la actualidad, gracias a esta herramienta, las empresas ya no tienen que hacer análisis manualmente. La inteligencia de negocios ayuda a que los procesos se hagan más rápido y fácil, lo cual beneficia de gran manera a las empresas que deciden incorporarlo (Microsoft, s.f.).

#### 2.4.1.2. *Elementos de Power BI*

Si un usuario desea trabajar con Power BI tiene 3 opciones diferentes, las cuales son:

- Power BI Desktop: Aplicación de escritorio de Windows que tiene todas las funcionalidades de Power BI

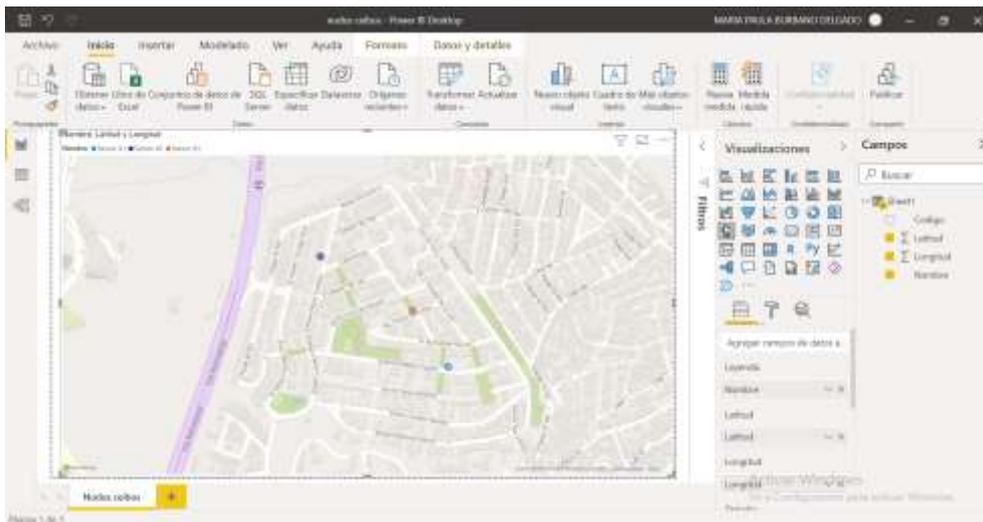


Figura 6. Power BI Desktop

- Servicio Power BI: Un servicio SaaS en línea. Se puede acceder a él buscándolo como powerbi.com.



Figura 7. Servicio Power BI

- Aplicaciones móviles de Power BI: está disponible para dispositivos Windows, iOS y Android (Microsoft, 2021).



Figura 8. Aplicación móvil de Power BI

Aparte de estos tres elementos, Power BI también presenta:

- Power BI Report Builder: herramienta útil para crear informes paginados que pueden ser publicados posteriormente en el servicio de Power BI. Un informe paginado es un tipo de informe en el que se especifica con que datos se trabaja (Microsoft, 2021).
- Power BI Report Server administra informes y KPI en un servidor de informes local con un portal web. (Microsoft, 2021).

Para el desarrollo de esta tesis se hará uso del servicio Power BI para la visualización grafica de la data que logre ser recogida por los sensores.

## **2.5. Elementos del hardware**

Anteriormente se explicó el funcionamiento del sensor que fue elegido para este proyecto. En esta sección se describirán las otras partes que se incorporaron al hardware para darle funcionamiento al sensor MQ135.

### **2.5.1. Arduino Nano**

Arduino Nano es una placa pequeña para desarrollo que tiene como base al microcontrolador ATmega328P. Posee 14 entradas/salidas digitales, 6 analógicas, y se puede conectar a un ordenador a través de una entrada de Mini-USB (Arduino, s.f.).

Los datos que se buscan captar con este tipo de placa a través de sensores normalmente son datos analógicos. Esto quiere decir que no solo existen dos posibles resultados, como en la temperatura, velocidad o en este caso en concreto el CO<sub>2</sub>. Esta placa cuenta con un conversor ADC, que significa que hace la conversión analógica digital de los datos captados del exterior.

En la figura 9 podemos observar una explicación gráfica de cómo funciona esta conversión. Lo que hace Arduino para llegar a leer señales analógicas es llevar a cabo aproximaciones discretas por medio de rectángulos digitales. Mientras más pequeño sea el rectángulo mayor semejanza tendrá con la señal analógica (Giraldo, s.f.).

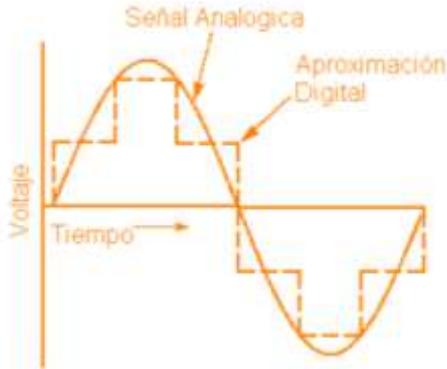


Figura 9. Conversión Analógica – Digital. Nota Tomado de: Control Automático Educación

Otra característica con la que cuenta el Arduino Nano es un protocolo llamado I2C el cual permite la comunicación interna entre dispositivos.

El protocolo I2C emplea un pin de reloj en serie (SCL) donde la placa de escritura Arduino examina intervalos regulares y un pin de datos en serie (SDA) por el cual se envía la información entre dispositivos.

Funciona transfiriendo un solo bit de información que le indica la dirección de un dispositivo y un comando o datos desde la placa al dispositivo I2C a través de la línea SDA. Después de enviado el bit, el dispositivo al que se le hizo la solicitud la ejecuta para posteriormente transmitir sus datos.

Para enviar datos desde Arduino se usan los pines SDA y SCL del mismo. Cada pulso en el pin SCL le indica al dispositivo que recibe los datos que lea el pin SDA. De esta forma los bits son puestos en SDA uno a la vez (Dannegård, 2021).

### 2.5.2. Cargadores de 5v 2A

Para el presente proyecto se utilizó un adaptador AC/DC que se encargara de la alimentación del dispositivo. Estos cargadores se conectan a un tomacorriente regular y convierte la corriente eléctrica de corriente alterna que recibe, a las necesidades del dispositivo que está conectado a la misma.

Hoy en día la mayoría de dispositivos que usamos hacen uso de este tipo de alimentación, como las laptops, celulares, entre muchos otros (Lee, s.f.).



*Figura 10. Cargador de 5V 2A. Nota Tomado de: Novatronic*

### 2.5.3. Convertidor TTL

Para el actual proyecto se necesitará un convertidor TTL, que tiene como propósito conectar dispositivos con diferente voltaje electrónico. En este caso se optó por un convertidor 5v a 3.3v, la conversión también es apta para darse en viceversa.

Gracias al convertidor se podrá generar una comunicación entre la placa Arduino Nano, que trabaja con 5v, y el Esp32 que funciona entre 2,55 y 3,6 voltios (Naylamp, s.f.). La descripción de la placa Esp32 se presentará a continuación.



Figura 11. Convertidor TTL. Nota Tomado de: Aliexpress

#### 2.5.4. Esp32

ESP32 es un chip con un microprocesador Tensilica Xtensa LX6. Proporciona Wi-Fi y en algunos casos conectividad Bluetooth de modo dual a dispositivos integrados.

La placa de desarrollo de este proyecto tiene preinstalado un chip ESP32 y se la denomina al igual que su chip. Estas placas sirven principalmente para crear prototipos de dispositivos IoT antes de llegar a producir en masa (Hübscmann, 2020).

La placa ESP32 cuenta, al igual que el Arduino Nano, con el protocolo I2C. Gracias a esto ambas placas podrán comunicarse entre sí y llevar a cabo el proceso de monitoreo del aire.



*Figura 12.* Placa ESP32. *Nota* Tomado de: CDMX Electrónica

### **2.5.5. Placa de circuito PCB**

Las placas de circuito impreso, también conocidas como PCB, se pueden definir como placas resistentes no conductoras construidas sobre una estructura basada en sustrato. Los PCB se utilizan principalmente para proporcionar conexión eléctrica y soporte mecánico a los componentes eléctricos de un circuito (Bhunia & Tehranipoor, 2019).

Una placa de circuito PCB es un material aislante plano y rígido que tiene estructuras conductoras delgadas adheridas a un lado. Las estructuras mencionadas crean patrones geométricos que consisten en, por ejemplo, rectángulos, círculos y cuadrados. Los rectángulos largos y delgados funcionan como interconexiones y varias formas funcionan como puntos de conexión para los componentes (Keim, 2020).

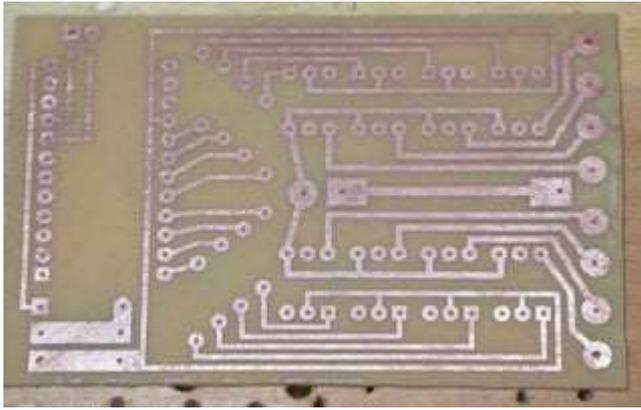


Figura 13. Placa de circuito PCB. Nota Tomado de: Instructables

### 2.5.6. Switch

Un switch es un dispositivo electromecánico que se utiliza para controlar la energía en los circuitos eléctricos y electrónicos encendiéndolo y apagándolo o también se denomina como hacer y romper un circuito eléctrico.

Cuando la corriente puede fluir de un contacto a otro, el interruptor está encendido. Por otro lado, cuando hay una separación que impide el flujo de corriente entre los contactos, el interruptor está apagado.

Para el presente trabajo de titulación se usará un interruptor basculante, como el que podemos observar en la figura, que se puede definir como un interruptor de encendido-apagado que oscila cuando se presiona el interruptor (Madhu, 2020).



Figura 14. Interruptor basculante. *Nota* Tomado de: Codrey Electronics

### **2.5.7. Cables jumper**

Existen tres tipos de cables que se usan frecuentemente para este tipo de proyectos. Los primeros y más utilizados son los cables macho-macho que cuentan en ambos de sus extremos con puntas metálicas que permiten que dos componentes estén conectados. Por otro lado, existen los cables macho-hembra y hembra macho, que su objetivo es alargar la distancia entre dos componentes. Para el uso de los dos últimos mencionados siempre se necesitará de cables macho-macho para que cumplan su función (Electrónica Caribe, s.f.).



Figura 15. Diferentes tipos de cables jumper. *Nota* Tomado de: Electrónica Caribe

## 2.6. Comunicación entre hardware, software y Power BI

El primer componente que hace parte de este sistema es el cargador 5v 2a, es el encargado de proporcionarle energía a los prototipos sin la cual los sensores no podrían funcionar. Para esto se incorporó un conector dentro de la caja del prototipo, que está conectado a 5v y a GND para su correcto funcionamiento.

Luego de que el dispositivo fue conectado a la corriente eléctrica, el sensor MQ135 empieza a funcionar, receptando datos del mundo exterior. Para que este sensor pueda transmitir esta información necesita estar conectado a alguna placa de desarrollo, en esta ocasión se escogió Arduino Nano.

Para realizar una lectura analógica de los datos debemos conectar la salida analógica del módulo del sensor, el pin A0, a un pin analógico de la placa Arduino. Adicionalmente, para lograr alimentar el sensor, debemos conectar el pin VCC del

MQ135 al pin de 5v de Arduino Nano. Similarmente, se deben conectar los pines GND de los dos dispositivos. Todas estas conexiones se pueden hacer a través de cables jumper.

A diferencia del sensor de gas, la placa Arduino nano cuenta con ocho pines analógicos, como se muestra en la figura 17, por lo que se puede conectar a cualquiera de estos (Santos, 2021).

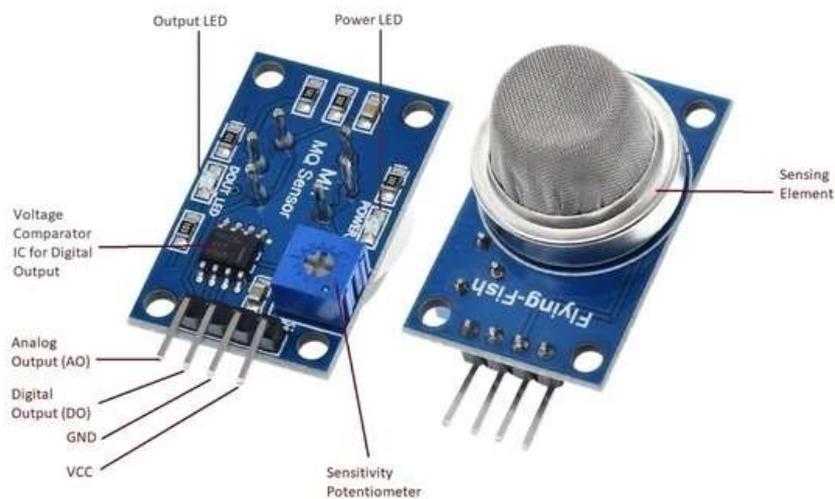


Figura 16. Partes del sensor MQ135. Nota Tomado de: Iberotecno

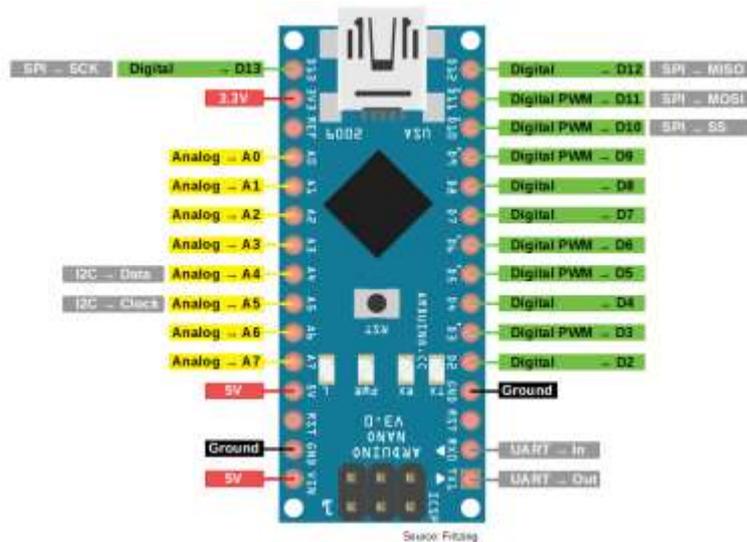


Figura 17. Partes de la placa Arduino Nano. Nota Tomado de: Roboteach

También es necesario leer el sensor desde el IDE Arduino y se puede lograr después de conversiones necesarias de los datos usando la librería “<MQUnifiedsensor.h>” y posteriormente el comando “readSensor()”.

```

void Sensor_MQ135(){
  MQ135.update(); // Update data, the arduino will be read the voltage on the analog pin
  // MQ135.setA(110.47); MQ135.setB(-2.862);
  CO2 = MQ135.readSensor();
  Serial.print("CO2: ");
  Serial.println(CO2);
  //delay(1000);
}

```

Figura 18. Comando readSensor dentro del código

Después de que leer los datos del sensor desde la placa Arduino, esta información debe ser transmitida hacia la placa ESP32. Para lograr esta comunicación en la parte del hardware se debe implementar un convertidor TTL ya que ambas placas cuentan con diferentes voltajes.

En la parte del hardware, es necesario conectar el pin SDA y SCL del Arduino Nano a los pines HV1 y HV2 que son de mayor voltaje. Luego de esto, se conectan los pines LV1 y LV2 (menor voltaje) del convertidor lógico TTL a los pines D27 y D26 del ESP32.

En la programación se envían los datos a través del protocolo I2C (anteriormente explicado) para finalmente ser enviados al servidor gracias a la programación dada en el programa del ESP32 (mirar anexos).

En la siguiente imagen se muestra como están conectados todos los dispositivos que pertenecen a la parte del hardware del presente trabajo, excluyendo a la pantalla OLED, ya que se presentaron problemas con la misma y no se pudo implementar en este proyecto.

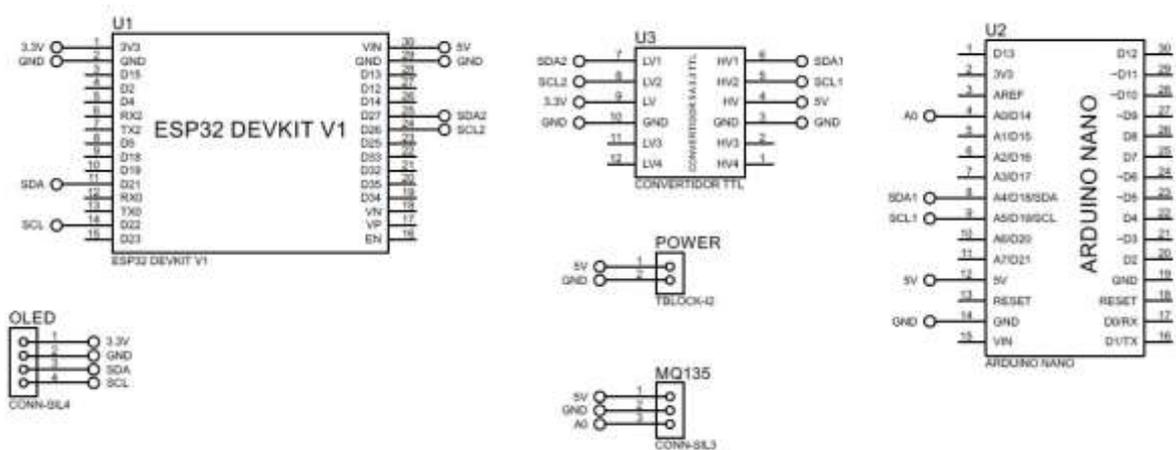


Figura 19. Conexiones del hardware de los prototipos.

Para enviar de datos se hizo uso del protocolo TCP, que significa protocolo de control de transmisión. TCP es una comunicación que sirve para enviar paquetes de datos a través de la red y asegura que la información se entregue al destinatario adecuado. TCP crea una conexión entre el nodo de origen y el de destino antes de transmitir los datos y mantiene viva la conexión hasta que la comunicación esté activa. En otras palabras, crea una conexión entre el origen y el destino posterior a la transmisión de datos y conserva la relación hasta que se encuentre activada la comunicación.

Esta comunicación funciona en base a una arquitectura cliente-servidor en donde el cliente es el que inicializa la interacción e implementa una relación con el servidor (Cerf, 1977).

Una vez que el servidor recibe esta información, la misma es interpretada y descompuesta para ser guardada en la base de datos de Aurora, la cual es una base de datos en SQL Server.

Para lograr enviar los datos desde la red Aurora hacia el Power BI, se hace uso de una puerta de enlace o mayormente conocido como Gateway. Un Gateway es un enrutador en una red informática, un punto de parada clave para los datos en su camino hacia o desde otras redes. Sin el Gateway el internet seria obsoleto, gracias a los mismos podemos comunicarnos y enviar datos de un lado a otro (Lewis, s.f.).

Ya teniendo acceso a la información necesaria, vamos a lograr visualizar la parte que nos interesa de la base de datos de Aurora en Power BI. Luego de esto, seleccionamos los gráficos que queremos que se muestren en Power BI y le

asignamos los valores a los campos correspondientes, como se muestra en la figura 21.

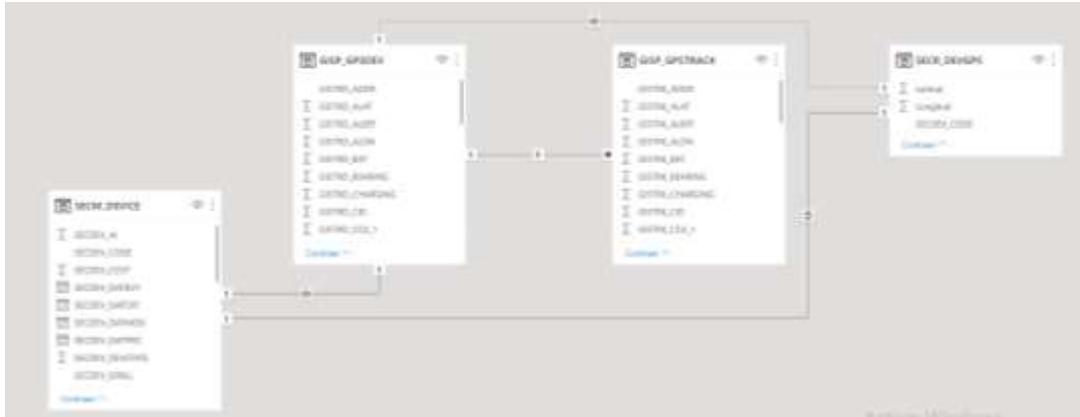


Figura 20. Base de datos disponible en Power BI.

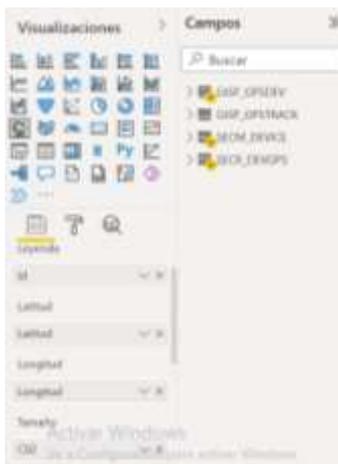


Figura 21. Campos utilizados para visualización del mapa.

## 2.7. Casos de uso

### 2.7.1. Monitoreo del aire para prevenir contagios de COVID-19

En la actualidad, el monitoreo del CO2 en el aire no solo se utiliza para reducir la contaminación. Los seres humanos al exhalar producimos CO2, pero adicionalmente expulsamos partículas que pueden ser portadoras de virus.

Debido a la reciente pandemia por COVID-19, muchas instituciones y países consideran importante reducir la cantidad de dióxido de carbono en espacios cerrados para reducir probabilidades de contagio.

Para evitar la transmisión del virus, el Gobierno italiano, y otros países europeos, adoptaron un plan de contingencia que incluye acciones capaces de garantizar una adecuada ventilación de aire en los salones de los colegios.

Empezaron el proyecto monitoreando en tiempo real de los niveles de CO<sub>2</sub> dentro de 9 escuelas ubicadas en el sur de Italia durante la reapertura de las escuelas. En concreto, realizaron actividades de seguimiento de datos para evaluar el escenario inicial en el interior de las aulas y las posibles mejoras obtenidas al aplicar un protocolo de ventilación, como abrir ventanas y puertas, basándose en los niveles de CO<sub>2</sub> obtenidos en el momento.

En la figura 22 podemos observar la diferencia entre los promedios del primer día de evaluación y el resto de días donde se llevaron a cabo los protocolos, siendo FD el primer día y OD los días con medidas.

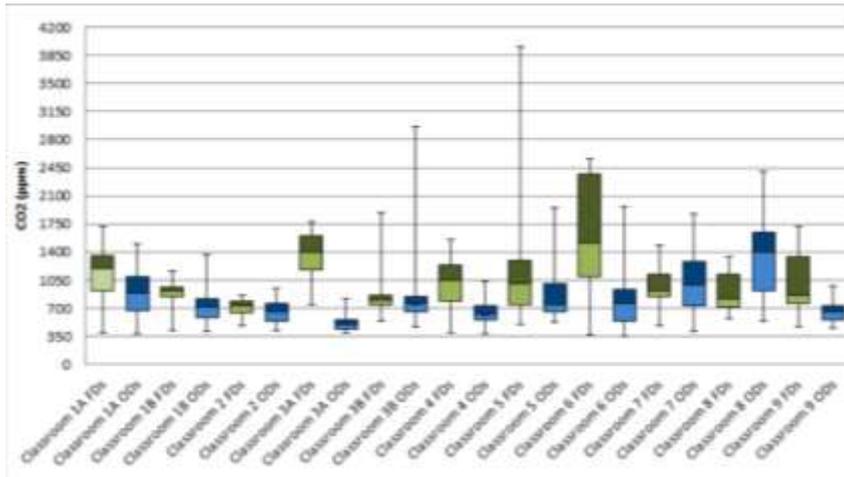


Figura 22. Variación de CO2 en el aire de las aulas. Nota Tomado de: Elsevier

Todas las aulas en las que los profesores cumplieron con la estrategia de ventilación demostraron una disminución en la concentración del gas anteriormente mencionado (Di Gilio, y otros, 2021).

Gracias a estas técnicas de control del CO2 en espacios cerrados, utilizando sensores con la misma base, pero más elaborados que los del presente trabajo, se puede disminuir efectivamente las posibilidades de contagios masivos por COVID-19.

### 2.7.2. Monitoreo del aire para exteriores

Al día de hoy existe una infraestructura global que surgió en Timisoara, Rumania llamada uRADMonitor. Se trata de dispositivos de seguimiento de la contaminación ambiental a través de sensores, con el fin de crear conciencia y mejorar la salud de las personas (Romanian Startups, s.f.).

El modelo A3 de uRADMonitor mide diferentes propiedades en el ambiente como la humedad, temperatura, presión barométrica, entre otras. Específicamente para el monitoreo de dióxido de carbono utiliza un sensor electroquímico de formaldehído, un sensor infrarrojo no dispersivo para medir la concentración de CO2 en el aire (Motisan, 2016).

A continuación, se muestra una captura de pantalla del sitio web de uRADMonitor, donde podemos observar todos los sensores que existen a nivel mundial en tiempo real. Como podemos darnos cuenta no existen en la actualidad este tipo de sensores en el Ecuador ni en sus países vecinos.

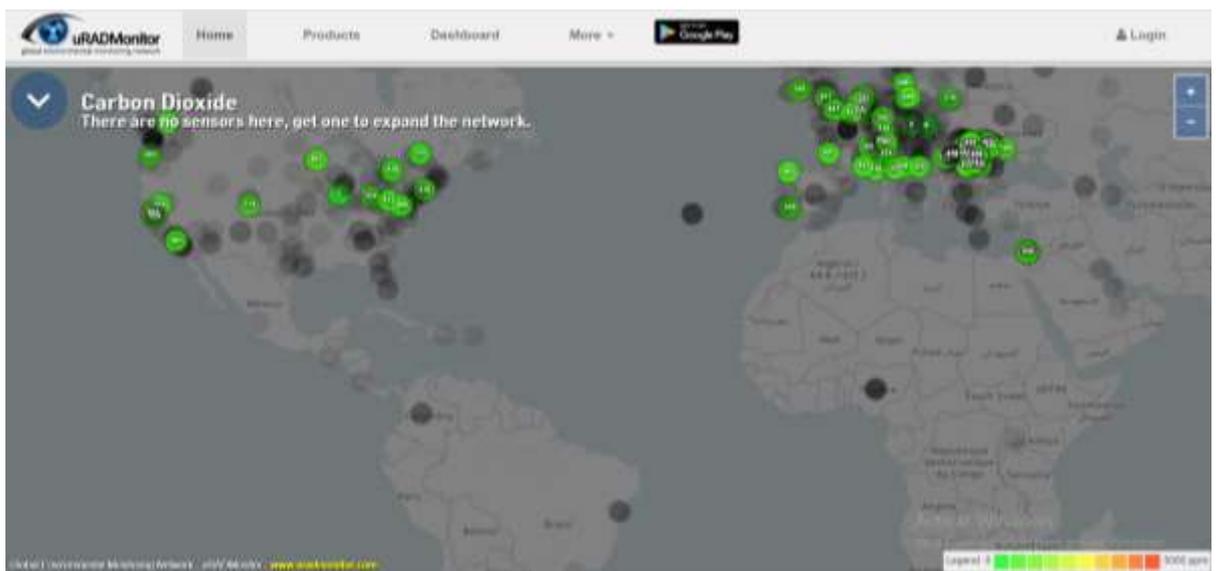


Figura 23. Monitoreo de CO2 en diferentes partes del mundo. Nota Tomado de: uRADMonitor

## **CAPITULO III**

### **METODOLOGÍA Y RESULTADOS**

El propósito de este capítulo es dar certeza de la utilidad del proyecto describiendo las técnicas que se utilizaron en los diferentes procesos. Se utilizarán metodologías para la parte de investigación, programación y para la parte de recolección de datos, buscando el fin de monitorear la calidad del aire de Ceibos Norte.

#### **3.1. Metodología de la investigación**

Las investigaciones científicas existen gracias a la curiosidad del hombre. Los humanos al ser los únicos seres dotados con inteligencia superior tenemos naturalmente una gran necesidad de saber el porqué de las cosas que nos rodean. Los descubrimientos del hombre empiezan en la prehistoria, y aunque algunos parezcan insignificantes cada uno de ellos nos ha guiado al camino para llegar a tener todas las cosas que tenemos hoy en día. Desde sofisticada tecnología hasta algo tan simple como un cepillo de dientes, es gracias a las investigaciones.

La investigación es algo de cada día en la actualidad. En todas las profesiones se necesita investigación para poder realizar tareas. Los abogados investigan los casos judiciales de sus clientes, los dentistas investigan la historia clínica dental de su paciente, y como estos muchos ejemplos más (Paz, 2000).

Para poder entender el significado de metodología de la investigación primero tenemos que entender el significado de método.

Podemos definir como método a un conjunto de técnicas que nos permiten lograr nuestras metas. En otras palabras, es una lista de pasos a seguir para alcanzar un objetivo. (Westreicher, 2020)

Por otro lado, la metodología se define como un conjunto de métodos de una investigación científica o la ciencia del método (Real Academia Española, 2021).

Existen 3 diferentes enfoques de la metodología de la investigación. A continuación, se definirá cada uno de ellos:

- Enfoque cualitativo: en este tipo de investigaciones se suele tener de base encuestas, entrevistas, puntos de vista. No utilizan mediciones numéricas y no es necesaria una prueba de hipótesis ya que buscan afinar las diferentes preguntas de la investigación. Su objetivo es comprender las variables involucradas en el proceso en lugar de medir y calcular. Es mayormente común ver este tipo de enfoque en proyectos sociales. Se suele criticar este método por ser muy especulativo.
- Enfoque cuantitativo: su fundamento principal son las mediciones numéricas y centran su investigación en los resultados de la recolección de estos datos. Recolecta, mide y da resúmenes estadísticos de la población en cuestión para definir si su hipótesis es correcta o no. El proceso de este enfoque empieza teniendo una idea de investigación. Luego, el segundo paso es definir la o las preguntas de investigación y

también los objetivos. A partir de esto se derivan las hipótesis y se procede a realizar los diferentes procesos necesarios de cálculo para poder contrastar la hipótesis. Es común ver el enfoque cuantitativo en investigaciones que puedan dar un resultado cuantificable.

- Enfoque mixto: En este enfoque se hace uso del enfoque cuantitativo y del cualitativo por igual. Se utilizan ambas técnicas para poder tener un resultado más preciso haciendo uso de las bondades de cada enfoque. Suelen hacer entrevistas, pedir opiniones, entre otras características cualitativas, pero también se busca valorar en escalas tangibles esta información, se observan las tendencias extraídas y se exponen las hipótesis (Cortés & Iglesias, 2004).

El presente trabajo de titulación tendrá una base mixta. Se buscará probar o rechazar la hipótesis anteriormente estipulada con los datos recolectados de los sensores de CO<sub>2</sub> y al mismo tiempo se llevará a cabo entrevistas a expertos en los campos de ambiente e internet de las cosas.

### **3.1.1. Recolección de datos**

Entre las herramientas de recolección de datos se optó por realizar entrevistas a expertos en los temas que se van a tratar.

La primera entrevista se la realiza al Ing. Diego Eraso Belalcázar. Actualmente trabaja en el campo de Ingeniería y Gestión Ambiental y cuenta con más de 15 años de experiencia en el campo.

Por otro lado, se quiso buscar un enfoque tecnológico para la recopilación de información por lo que se entrevistó al dueño del servidor Aurora, el cual presta sus servicios para el desarrollo de este proyecto. El Ing. Giancarlo Molina se graduó en la Universidad Católica de Santiago de Guayaquil con el título de Ingeniero en Sistema y lleva 10 años trabajando en el área de internet de las cosas.

## **3.2. Metodología de desarrollo**

Para el presente proyecto se las metodologías de desarrollo más adecuadas son la metodología de desarrollo por prototipos y la metodología de sistema embebido.

### **3.2.1. Metodología de desarrollo por prototipos**

Esta metodología es la más adecuada para el actual proyecto porque precisamente se llevará a cabo la implementación de tres prototipos de sensores. Las etapas para efectuar la metodología en cuestión serán mencionadas a continuación:

#### *3.2.1.1. Recolección de requisitos.*

Para esta fase se tomó en cuenta la carencia de una solución IoT para el monitoreo del aire en la ciudadela Ceibos Norte. A partir de estos hechos se realizaron una entrevista a un Ingeniero Ambiental y a un Ingeniero en sistemas que lleva años trabajando en el campo de IoT.

En Guayaquil existe actualmente un sistema de monitoreo del aire gracias al municipio de la ciudad que actualmente se ubica en 23 puntos diferentes (El Universo,

2021). Aunque es una buena alternativa no cubre todos los puntos de la ciudad y los datos solo son de fácil acceso para las autoridades. El prototipo del actual proyecto se presenta como una alternativa más económica y más accesible a las ya existentes.

### *3.2.1.2. Diseño Rápido.*

Esta fase estará enfocada en la arquitectura del prototipo, es decir como esta interconectada y de qué manera funciona cada uno de los elementos implicados. Se describirá cómo funciona la arquitectura de la conexión a Power BI, del hardware y del software.

La primera arquitectura que se describirá será la de la comunicación hacia Power BI. Hay 4 capas que podemos usar de base, la capa objetos en la que están presentes los 3 sensores que monitorean la cantidad de CO<sub>2</sub> en el aire. Estos sensores se conectan a través de una red wifi a el servidor Aurora, donde se almacenarán los datos adquiridos por los sensores. Finalmente, en la última capa llamada capa de aplicación, se muestran los datos gráficamente en la herramienta seleccionada posteriormente, la cual es Power BI.

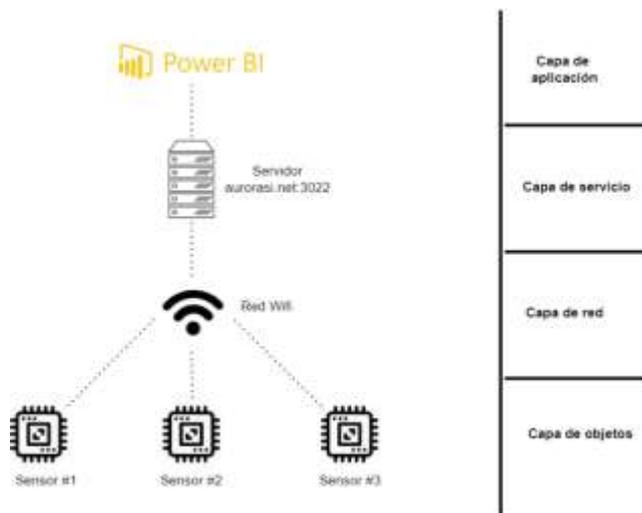


Figura 24. Arquitectura de la comunicación IoT hasta Power BI

Por otro lado, tenemos la arquitectura del hardware que consta de 4 elementos importantes como podemos apreciar en la figura 25. A continuación, se explicara cómo funcionan los dispositivos y cada una de sus piezas.

El sensor MQ135 se conecta al Arduino Nano por medio de ADC (convertidor analógico-digital) que hace su proceso con el firmware que es el programa cargado al Arduino.

Entre la comunicación del Arduino Nano y el ESP32 se incorporó un convertidor TTL lógico para evitar conflictos en la comunicación entre ambos. Ya que el primero respectivamente funciona con 5 volteos, y el segundo con 3.3 volteos.

Los I2C de las placas respectivas trabajan muy similar a un servidor-cliente. Ya que a través de estos el ESP32 envía una solicitud al Arduino Nano y este responde con los datos que fueron programados anteriormente.

Al ya recibir los datos a través del I2C del ESP32, el firmware los procesa y continua a hacer los procedimientos necesarios con el wifi para posteriormente ser enviados al servidor.

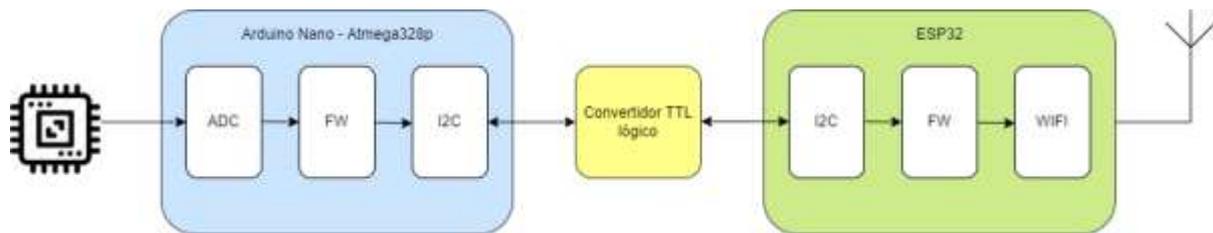


Figura 25. Arquitectura del hardware

### 3.2.1.3. Fabricación de prototipos

En la actual etapa se llevó a cabo la fabricación de los tres prototipos que se basó en 2 etapas: desarrollo del software en Arduino y construcción del hardware. La fabricación de ambas etapas se realizó a partir de las arquitecturas señaladas en el punto anterior.

En la figura 26 y 27 podemos observar uno de los prototipos ya armado. Cuenta con una caja protectora que fue construida en impresión 3D, con el fin de evitar contratiempos por lluvia o suciedad.

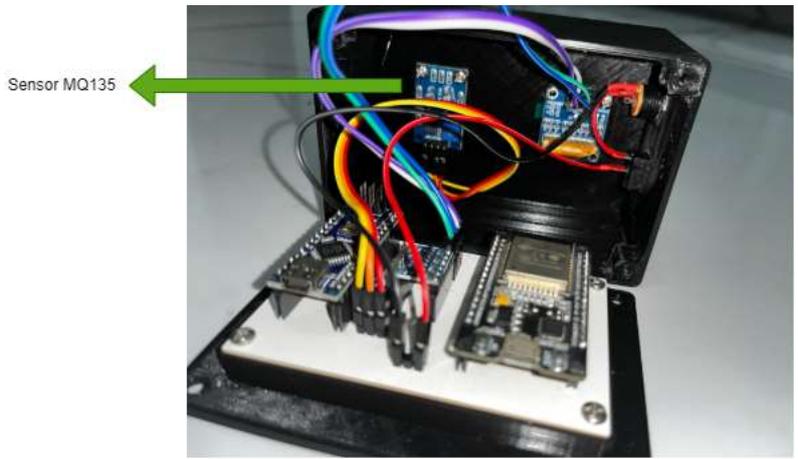


Figura 26. Parte superior del prototipo IoT

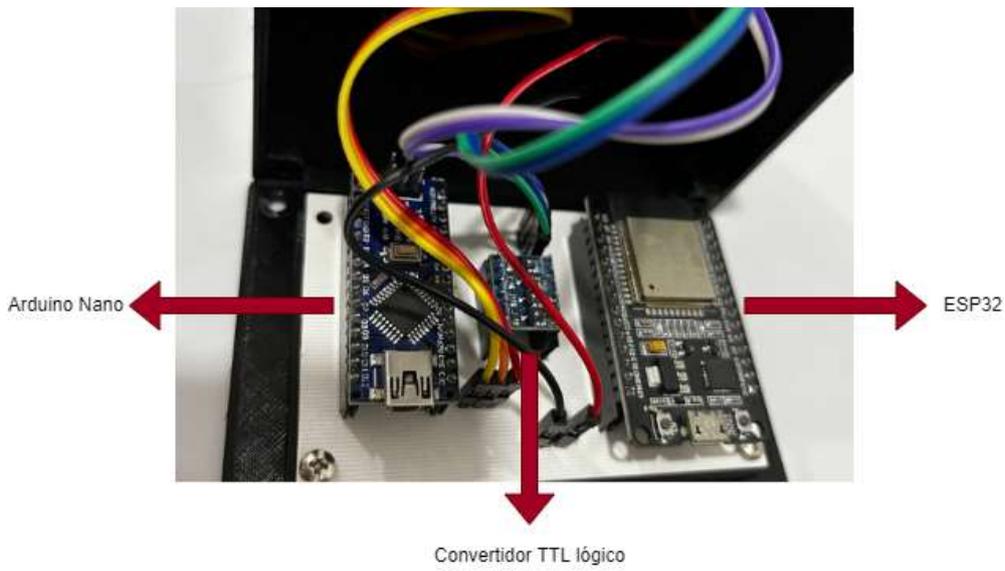


Figura 27. Parte inferior del prototipo IoT

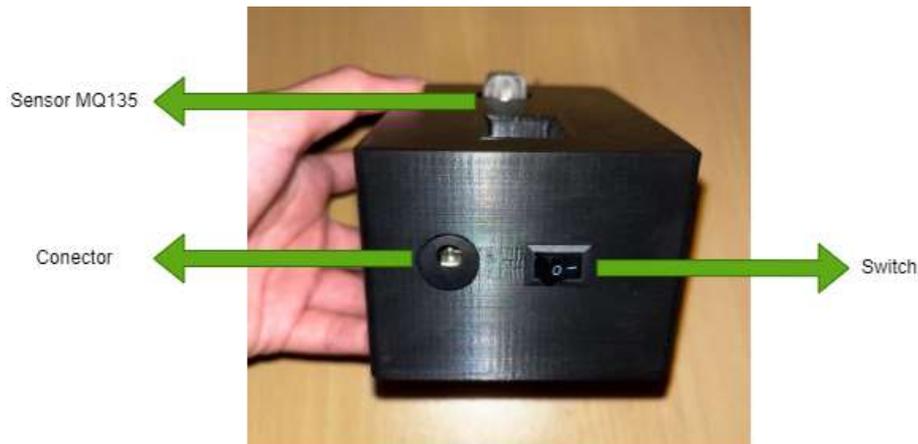


Figura 28. Exterior del prototipo.

En la parte del software se crearon dos programas para la comunicación entre las diferentes placas. En la figura 29 se muestra una parte del código del Arduino Nano, que se encarga de mandar los datos al convertidor y posteriormente al ESP32.

```

I2c_esclavo_arduino_MQ135 Arduino 1.8.19
Archivo Editar Programa Herramientas Ayuda

I2c_esclavo_arduino_MQ135
// ***** DATA ARDUINO *****
#include <MQ135UnifiedSensor.h>
#include <SoftwareSerial.h>
#include <Wire.h>

String message;
bool B = false;

String header = "I2c:";

#define placa "Arduino Uno"
#define Voltage_Resolution 5
#define pin AQ //Analog input 0 of your arduino
#define type "MQ-135" //MQ135
#define ADC_BIT_RESOLUTION 10 // For arduino UNO/MEGA/NANO
#define RatioMQ135CleanAir 3.6//10 / 10 = 3.6 ppm

double CO2 = 0;

MQ135UnifiedSensor MQ135(placa, Voltage_Resolution, ADC_BIT_RESOLUTION, pin, type);

void MQ135_setup()
//Set each model to calculate the PPM concentration and the value of constants
MQ135.setRegressionMethod(1); // PPM = a*ratio^b
MQ135.setA(110.47); MQ135.setB(-2.962); // Configure the equation values to get CO2 concentration

```

Figura 29. Parte del código para la placa Arduino Nano

Se creo otro programa del mencionado anteriormente para la placa ESP32, que hace peticiones al Arduino Nano y envía todos los datos al servidor de Aurora, mediante la configuración que tiene de puerto y host que es la IP publica de Aurora.



```
i2c_maestro_esp32 Arduino 1.8.19
Archivo Editar Programa Herramientas Ayuda

// CÓDIGO PARA ESP32
#include <Gpio.h>
#include <OLEDB_I2C.h>
#include <WiFi.h>

//PRIMER PUERTO I2C por defecto
#define I2C_SDA 21
#define I2C_SCL 22

// SEGUNDO PUERTO I2C
#define I2C_SDA2 27
#define I2C_SCL2 26

TwoWire I2C = TwoWire(1);
//-----//
String Nombre_Equipo = "A22"; // Nombre de equipo o prototipo
const char* ssid = "D-LINK_OIE-615"; // Nombre de la red wifi
const char* password = "9713381109463"; // Contraseña de la red wifi
//-----//
const uint16_t port = 30022;
const char * host = "auroraxi.net";
WiFiClient client;
String IoTdata;
String startbit = "#IOT#";
```

Figura 30. Parte del código para la placa ESP32

Para finalizar con la parte del software de este proyecto, en la próxima figura se puede observar a través del monitor serie los datos capturados cada 5 segundos y adicionalmente él envió de data al servidor en un intervalo de un minuto.

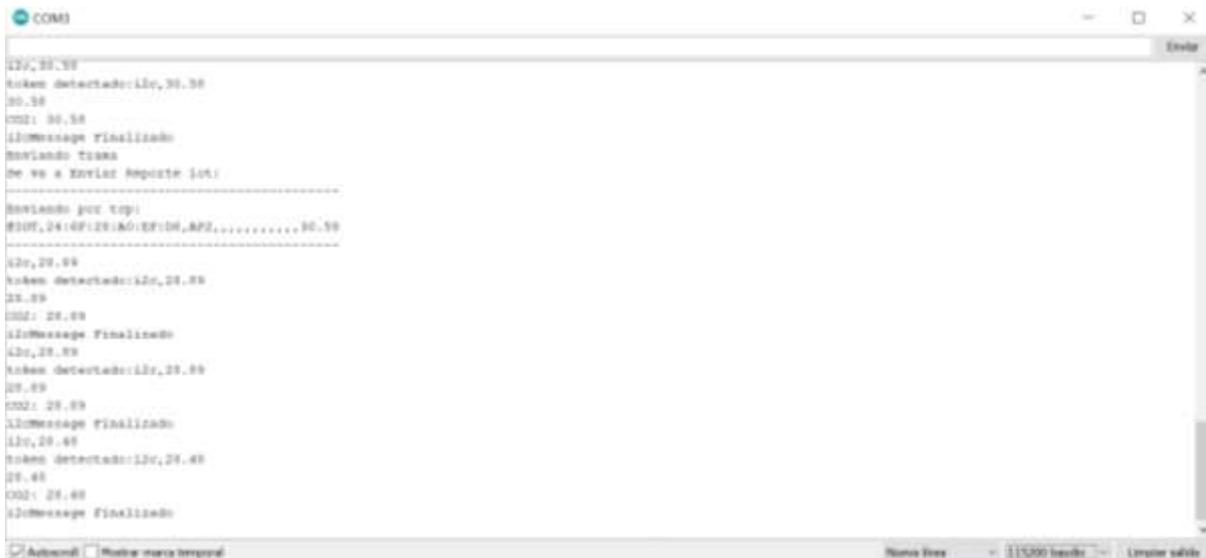


Figura 31. Datos en el monitor serie del IDE Arduino

#### 3.2.1.4. Evaluación de prototipos

En esta etapa los prototipos fueron evaluados por el tutor del presente proyecto, el Ing. Colon Celleri. Para la revisión de este proyecto la base fueron los objetivos y el alcance del proyecto propuestos en los inicios del semestre.

Se realizaron ciertas modificaciones a nivel del programa que mostraría los datos, ya que se llegó a la conclusión que Power BI es una herramienta mucho más eficiente cuando se trata de representación gráfica de información.

A nivel de hardware no se consideró necesario ninguna modificación ya que el prototipo final logro cumplir las expectativas y llevar de manera correcta la información al servidor para luego ser enviado al Power BI.

### 3.2.1.5. Producto final

Luego de proceder a los cambios anteriormente mencionados, se da como terminada la fase de desarrollo del proyecto.

En la figura 32 podemos observar uno de los prototipos conectado a una corriente eléctrica listo para usarse y con el programa cargado anteriormente.



Figura 32. Prototipo conectado a enchufe eléctrico

A continuación, se presentan capturas del log in de la interfaz habilitada para mi persona con el fin de poder observar los datos que llegan al servidor en tiempo real.



Figura 33. Ingreso al portal de servicios de Aurora

Después de ingresar, se puede observar a los tres sensores activos con los nombres de ap1, ap2 y ap3. Adicionalmente se puede consultar la hora en la que se captaron los datos, la fecha, la dirección MAC del prototipo y la cantidad de ppm de CO2 que se captó en ese momento.

		240AC4322E	AP3	240AC4322ED4	HT	---	2022/01/19	0	0	-1	-1	-1	-1	1.74
					HT	---	16:38:02							
		246F28A0EF	ap2	246F28A0EFD8	HT	---	2022/01/19	0	0	-1	-1	-1	-1	73.65
					HT	---	14:52:52							
		K101	AP1	94B97E1B2154	HT	---	2022/01/12	0	0	-1	-1	-1	-1	0.27
					HT	---	12:01:06							

Figura 34. Prototipos A1, A2, A3 en el portal de servicios de Aurora

Adicionalmente si se desea observar detalladamente solo uno de los sensores se puede seleccionar la opción detalles del evento. En esta nueva ventana se mostrará la cantidad de CO2 en los diferentes minutos que el prototipo ha estado conectado, es decir, su historial.

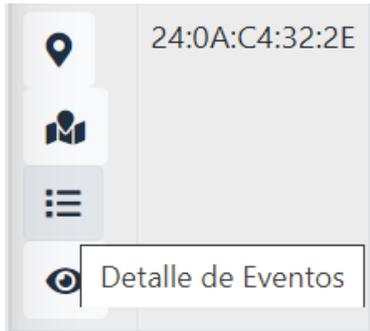


Figura 35. Detalle de eventos en Aurora

Fecha	id	co2	wif	wss	Sec	ip	Puerto
2022-01-20 18:02:11	0	1149.40	---	---	100	192.168.41.100	3002
2022-01-20 18:03:54	0	35.27	---	---	100	192.168.41.100	3002
2022-01-20 18:04:34	1	27.1	---	---	100	192.168.41.100	3002
2022-01-20 17:06:56	0	1.83	---	---	100	192.168.41.100	3002
2022-01-20 17:17:36	1	0.96	---	---	100	192.168.41.100	3002
2022-01-20 17:18:56	2	3.54	---	---	100	192.168.41.100	3002
2022-01-20 17:19:56	3	0.47	---	---	100	192.168.41.100	3002
2022-01-20 17:20:56	4	0.34	---	---	100	192.168.41.100	3002
2022-01-20 17:21:57	5	0.34	---	---	100	192.168.41.100	3002
2022-01-20 17:22:57	6	0.19	---	---	100	192.168.41.100	3002

Figura 36. Historial del sensor AP3

Para lograr una visualización de los datos más placentera, se optó por la herramienta Power BI para graficar las variantes de CO2. A continuación, se muestra las pantallas creadas en la página de servicios anteriormente mencionada donde podemos observar los tres puntos en el mapa, y los valores de CO2 de cada uno. Adicionalmente, se creó otra pantalla donde podemos observar diferentes graficas que representan el CO2 en cada uno de los puntos a través del tiempo.

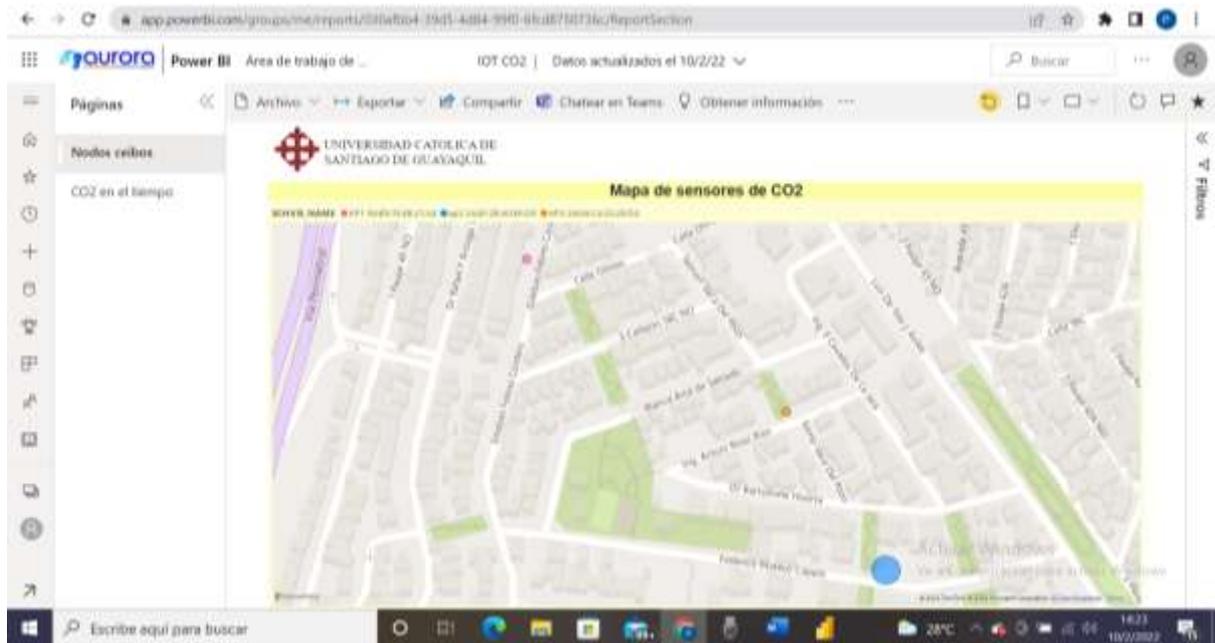


Figura 37. Mapa de Ceibos Norte con los puntos de los prototipos señalados

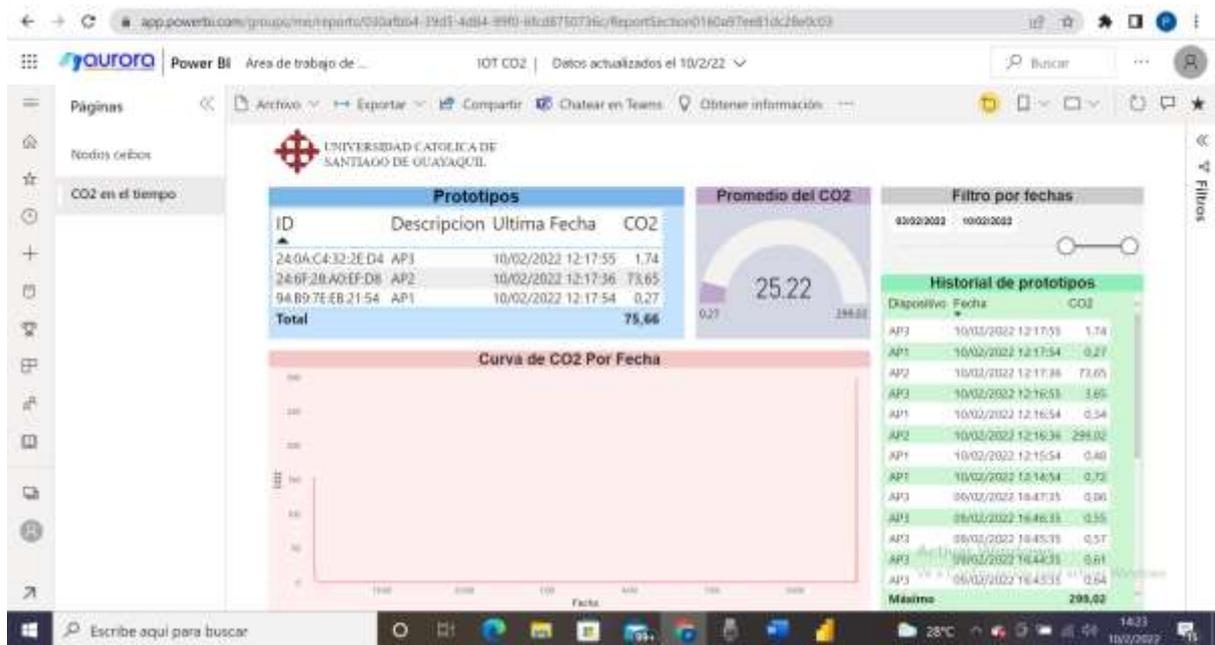


Figura 38. Gráficos del CO2 a través del tiempo

### 3.2.2. Metodología de sistema embebido

La metodología de sistema embebidos se basa en la descripción modelo y modelado del hardware y software que se utilizan en el prototipo el cual es utilizado

como una alternativa o solución IoT para medir CO2 dentro de una urbanización usando una red wifi de banda de 2.4GHz.

Se han utilizado dos placas diferentes que posteriormente han sido programadas por el IDE de Arduino con el fin de obtener datos en tiempo real de la concentración de CO2 en diferentes puntos de Ceibos Norte. Una de ellas se denomina ESP32, una placa de alta confiabilidad en su conexión a redes wifi, y a su vez la tarjeta Arduino nano con un microcontrolador Atmega328p que adquiere datos del sensor MQ135.



Figura 39. Metodología de sistema embebido

### 3.3. Análisis de resultados

Para la recolección de datos en el presente trabajo se usaron diferentes métodos como las encuestas y entrevistas, las cuales ayudaron a la implementación de la solución IoT empleando el servidor de Aurora.

### **3.3.1. Análisis de las entrevistas**

#### *3.3.1.1. Entrevista con Ing. Ambiental*

La entrevista con el Ing. Diego Eraso se realizó a través de correo electrónico y se pudo concluir que, aunque el CO<sub>2</sub> no es un gas que tengas límites establecidos por el Acuerdo Ministerial 097-A/ 2015 a diferencia de otros gases, si se considera que su presencia es un problema en la actualidad. Esto se debe a que los humanos lo generamos de manera excesiva, como consecuencia contamina la atmósfera y pone en peligro nuestra salud.

Por otro lado, el Ingeniero pudo concluir que hay varias cosas que podemos hacer para reducir nuestro impacto en esta problemática. Primeramente, explico que lo más importante es concientizar a las personas, hacerles saber sobre las consecuencias irreversibles que esto puede tener en nuestro planeta y como por ende esto afecta nuestra salud. Entre otras acciones de cambio que podemos realizar se señaló el reducir el uso de carro privado, reducir nuestro gasto energético y plantar árboles.

El entrevistado determino que el gobierno ecuatoriano también debe tomar acción, siendo una de las posibilidades basarse en el “Acuerdo de París”. El acuerdo mencionado busca transformar económica y socialmente diferentes países mejorando la resiliencia al cambio climático. Se trata principalmente de un tratado internacional sobre el cambio climático y tiene como objetivo acortar el cambio climático.

Finalmente, el último tema del que se discutió en la entrevista fue sobre la importancia de tomar acciones para frenar el cambio climático en el Ecuador. De este tema se dedujo que es muy importante reducir la emisión de gases de efecto invernadero, ya que esto ayuda a preservar el planeta. Se comentó también que en el país si se cuenta con una estrategia que se está implementando, pero no ha dado frutos concretos.



Figura 40. Entrevista al Ing. Diego Eraso vía e-mail.

### 3.3.1.2. Entrevista con experto en IoT

La entrevista con el Ing. Giancarlo Molina tuvo un enfoque totalmente diferente, se enfocó principalmente en el internet de las cosas, en cómo afectan nuestra vida y como puede ser la solución a muchos problemas en la actualidad, incluyendo problemas ambientales.

Después de realizar la entrevista podemos concluir que el internet de las cosas ya está cambiando nuestras vidas y que en el futuro se espera ver muchos más avances relacionados con el tema, como los carros inteligentes, lo cual nos ofrecerá más confort pero que al mismo tiempo afectaran la privacidad y seguridad. Dado esto

no solo tiene que haber un avance en el internet de las cosas si no también en como protegemos nuestros datos ante posibles ataques.

En la entrevista también se habló de diferentes formas que se está incluyendo el IoT en la conservación ambiental, se mencionó el término “Smart Environment” que se trata del monitoreo de la calidad del agua, el aire y también la energía. Se concluyo que la tecnología es una gran alternativa no solo para nuestra propia comodidad sino también para cuidar el planeta.

Por otro lado, también se habló específicamente del monitoreo del aire por medio de soluciones IoT, por lo que se llegó a la conclusión de es una buena manera de disminuir la contaminación especialmente en áreas muy pobladas. Gracias a los monitoreos se puede analizar de manera histórica en busca de patrones de comportamiento además de predicciones y simulaciones.

Para finalizar la entrevista, se habló de la relación costo/beneficio del IoT. Se pudo llegar a la conclusión de que esto depende de la magnitud del proyecto. Hay pequeños proyectos como el del actual trabajo que no requieren de mucha inversión para funcionar, pero si se piensan en proyectos masivos si se requiere invertir grandes cantidades de dinero. Es por esto que se tiene que tomar en cuenta la importancia del proyecto, que efectos positivos pueden tener y lo que se tiene pensado invertir.



Figura 41. Entrevista con el Ing. Giancarlo Molina por videollamada

### **3.3.2. Análisis de la calidad del aire por medio de la solución IoT implementada**

Después de realizar pruebas en los diferentes puntos en la ciudadela podemos sacar las siguientes conclusiones:

- Los niveles de CO<sub>2</sub> en el punto AP1 tienen el promedio más bajo. El valor más alto que dio fue de 12,52 ppm, que sigue siendo un valor muy bajo de CO<sub>2</sub> por lo que podemos decir que la zona tiene un aire bastante limpio.
- En el prototipo AP2, podemos observar un valor máximo de 1913 ppm, lo cual sobrepasa los niveles recomendados de 400 ppm mencionados al principio de este documento. Adicionalmente, su promedio se mantiene bastante alto comparado a los demás. Al ser la zona que esta más cercana a la Vía

Perimetral podemos concluir gracias a los datos que es la que más riesgo tiene de aire contaminado.

- En el último prototipo, el AP3 que se encuentra en la mitad de la ciudadela, también se detectaron niveles altos de CO2. Llegando hasta 1149ppm. Siendo un poco menor a los niveles del AP2, pero igualmente mayor a lo que recomienda. A pesar de esto, su promedio se mantiene bastante bajo.

Cabe recalcar que los niveles altos dados se pueden dar por la calibración que los sensores necesitan antes de funcionar correctamente. Teniendo esto en cuenta podemos centrarnos en los promedios que nos arrojan los datos, en los cuales todos los sensores presentan datos estables y bajo los límites permitidos de CO2, por lo que podemos concluir que la ciudadela de Ceibos Norte tiene un aire seguro para sus habitantes.

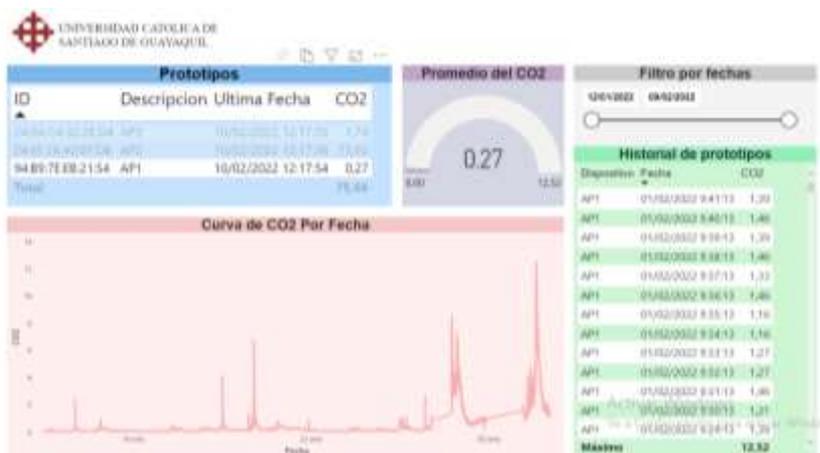


Figura 42. Resultados del prototipo AP1



Figura 43. Resultados del prototipo AP2



Figura 44. Resultados del prototipo AP3

### 3.3.3. Análisis de la situación actual del monitoreo del aire en Guayaquil

Desde el año 2020 la Municipalidad de Guayaquil ha contratado un servicio de monitoreo de la calidad del aire en 23 puntos diferentes dentro de la ciudad, su plan es contratarlo por 2 años y tendrá un costo de \$159,36 anual.

Vía a la Costa es uno de los puntos donde se lleva a cabo este monitoreo y gracias a los datos recolectados se pudo concluir que se está dentro de los límites permisibles. También se midieron los niveles en sectores como Mall del Sur, San Marino, Malecón 2000 y todos regresaron con niveles normales. Cabe destacar que

las pruebas se realizaron en los meses de restricción vehicular debido a la pandemia por covid-19, por esta razón los datos pudieron verse alterados.

Gracias a estos estudios también se pudo confirmar que la hora donde más se elevaban los niveles era a las seis de la tarde y después de las ocho de la noche la congestión vehicular disminuía considerablemente y por ende también la contaminación. Hasta el siguiente día a partir de las nueve de la mañana que vuelven a subir (El Universo, 2021).

El último programa de la calidad del aire se dio en el 2009 por el MAE, pero no se lograron encontrar datos de dicho proyecto simplemente los objetivos que buscaban alcanzar en aquella época (MAE, s.f.).

Estos son los datos más actualizados con respecto a la calidad del aire en la ciudad de Guayaquil, actualmente no hay ningún monitoreo constante con el que podamos analizar los diferentes niveles.

## **CAPITULO IV**

### **PROPUESTA**

En este último capítulo del trabajo de titulación se realizará una propuesta de internet de las cosas en la ciudadela Ceibos Norte, con el fin de cuidar de la salud de sus habitantes y preservar el medio ambiente que lo rodea. Teniendo el conocimiento previo de la falta de un monitoreo continuo de la calidad del aire en Guayaquil en la actualidad, se comunica sobre la alternativa IoT planteada en este proyecto vinculada a la red de Aurora y que cumple con los objetivos decretados al principio del presente documento.

#### **4.1. Objetivo**

Conocer la actual situación de la calidad del aire en Ceibos Norte a través de los 3 prototipos diseñados y de esta manera tener la seguridad de que los que habitan la ciudadela no están en riesgo de perjudicar su salud.

#### **4.2. Responsable**

Para el presente trabajo de monitoreo no se necesitó la autorización del comité o el presidente de Ceibos Norte, ya que es un proyecto ajeno al mismo y los sensores serán colocados en casas privadas de los propietarios que residen actualmente en la urbanización.

Fue necesario el apoyo del Ing. Molina para la realización de este proyecto, el cual brindo sus servicios de almacenamiento de los datos que fueron enviados al servidor Aurora.

### **4.3. Descripción de la solución IoT**

La solución IoT a llevar a cabo en la ciudadela Ceibos Norte para medir la calidad del aire en 3 diferentes puntos, los cuales serán: mi domicilio que se encuentra en la calle Fresnos, el domicilio de un familiar en la calle Abetos, así como, en la de mi entrenador ubicada en la calle Juncos.

Esta solución está compuesta por prototipos que contienen cada uno una placa Arduino y una ESP32, las cuales envían los datos al servidor Aurora y se guarda en su base de datos, para posteriormente mostrarse en mi Power BI.

### **4.4. Herramientas tecnológicas**

A continuación, serán mencionadas las herramientas que se necesitaron para la fabricación de los prototipos de monitoreo del CO<sub>2</sub> y el correcto funcionamiento del hardware.

- Sensor MQ135: su función es captar la cantidad de ppm del CO<sub>2</sub> de su alrededor.
- Arduino Nano: la placa Arduino nano es necesaria para la comunicación con el ESP32 y captar la información mandada por los sensores.
- ESP32: se encargará de enviar los datos a la red seleccionada.
- Convertidor TTL: se necesita un convertidor para la comunicación entre el Arduino nano y el ESP32, ya que funcionan con diferentes cargas electrónicas.
- Cargadores de 5V: su función será permitir conectar el prototipo a una corriente electrónica para que reciba energía.

- Cajas de prototipos 3D: el propósito de las cajas 3D en este proyecto serán proteger los prototipos de posibles lluvias o suciedad.
- Cables jumper: los cables jumper tienen el objetivo de conectar los diferentes dispositivos que forman el prototipo en su totalidad.
- IDE Arduino: es el más usado para este tipo de proyecto y fue necesario para programar las placas y sus microcontroladores.
- Red Aurora: a pesar que la red suele tener un precio de servicio, fue gratuito para este proyecto y por la misma razón fue la primera opción.

Para el correcto funcionamiento de la interfaz gráfica se necesitó Power BI Pro en línea, para tener la capacidad de actualizar datos automáticamente.

#### **4.5. Proceso**

Las fuentes de contaminación más fuertes de la ciudadela Ceibos Norte son los carros particulares de los residentes y la cercanía que mantiene con la Vía Perimetral por la cual circulan grandes cantidades de automóviles, motos y camiones la mayor parte del día.

Para seleccionar la ubicación de los sensores, se consideró necesario una distancia prudente entre cada prototipo para poder determinar qué sector de la ciudadela se considera más contaminante. Por otra parte, se optó por domicilios de conocidos de la autora del presente trabajo de titulación. La sección específica de cada casa se escogió de acuerdo a la cercanía a la calle, sin dejar de lado la necesidad de una red wifi cercana y enchufes eléctricos.

El primer punto (AP1) que se selecciono fue el de mi domicilio, en la calle Fresnos, el cual sé que se encuentra más distanciado de la Vía perimetral.



Figura 45. Punto de ubicación del sensor AP1.

El segundo prototipo (AP2) se situará en la calle Juncos que se encuentra más cercana a la Vía Perimetral.



Figura 46. Punto de ubicación del sensor AP2.

Por último, el tercer sensor (AP3) se piensa ubicar en la calle Isoras, la cual se encuentra en un punto medio en la urbanización.



*Figura 47.* Punto de ubicación del sensor AP3.

El segundo paso fue instalar el hardware en los puntos anteriormente seleccionados.



*Figura 48. AP1 instalado*



*Figura 49. AP2 instalado*



*Figura 50. AP3 instalado*

Posterior a la instalación de los prototipos se lleva a cabo la comunicación entre cada uno de los sensores con la red Aurora a través de los procesos mencionados a continuación:

- El sensor MQ135 percibe la cantidad de CO<sub>2</sub> en el ambiente por medio de los cambios de voltaje que se generan.
- La placa Arduino toma este voltaje y lo envía al ESP32 a través de un convertidor TTL.
- Por medio de la configuración wifi en el firmware, el ESP32 manda esta información hacia el servidor Aurora.
- El servidor Aurora recepta esta información por medio de un software interpretador, como se muestra en la siguiente imagen.

```

0.0.0.0:30022 ver 3.4 (build 2) 2022-02-09 13:28:02 - 2022-02-09 16:31:34
- 2022-02-09 13:28:02 Starting Program.
- Un sola instancia de D:\parcer\30022\TcpListenerApp.exe
- 2022-02-09 13:28:02 Starting listener...
- 2022-02-09 13:28:02 Hour zone:-5
- 2022-02-09 13:28:02 Ciclo maximo por hilo:0
- 2022-02-09 13:28:02 Vervose:True
- 2022-02-09 13:28:02 Process Responce:True
- 2022-02-09 13:28:02 Save On DB:True
- 2022-02-09 13:28:02 logpath:D:\parcer\data\log
- 2022-02-09 13:28:02 Errorlogpath:d:\parcer\errorlog
- 2022-02-09 13:28:02 NoProcesslogpath:D:\parcer\NoProcesslog
- 2022-02-09 13:28:02 Address Server:http://192.168.1.26/osm/osm.ashx?
- 2022-02-09 13:28:02 Segundos para reiniciar:24000
- 2022-02-09 13:28:02 Listening on 30022
- 2022-02-09 16:31:34 Received connection request from 190.131.31.92
- 2022-02-09 16:32:34 - received [43 bytes]
- Model:AURORAIOT,DeviceId:24:0A:C4:32:2E:D4,UTC Date:2022-02-09 21:32:34,Longitude:0,Latitude:0,Speed:0,
  lite:0,Report Type:,A Input 1:0,A Input 2:0,co2 sensor 1:152,74
- 2022-02-09 16:32:34 HI AURORAIOT 24:0A:C4:32:2E:D4 : 190.131.31.92 Threads: 1
- 2022-02-09 16:33:34 - received [42 bytes]
- Model:AURORAIOT,DeviceId:24:0A:C4:32:2E:D4,UTC Date:2022-02-09 21:33:34,Longitude:0,Latitude:0,Speed:0,
  lite:0,Report Type:,A Input 1:0,A Input 2:0,co2 sensor 1:20,67

```

Figura 51. Software interpretador de Aurora

- Los datos se guardan en la base de datos de SQL Server de Aurora.

SECDEV_CODE	GISTRD_DATE	GISTRD_SAT	GISTRD_CO2_1	GISTRD_DATMOD
24:0A:C4:32:2E:D4	2022-01-19 16:08:02.400	0	1.2	2022-02-09 16:39:34.937

Figura 52. Base de datos SQL Server de Aurora

- Posteriormente se envían los datos al portal de servicios donde puede ser visualizada por los usuarios que tienen acceso. Al mismo tiempo se mandan a través de Gateway a Power BI.

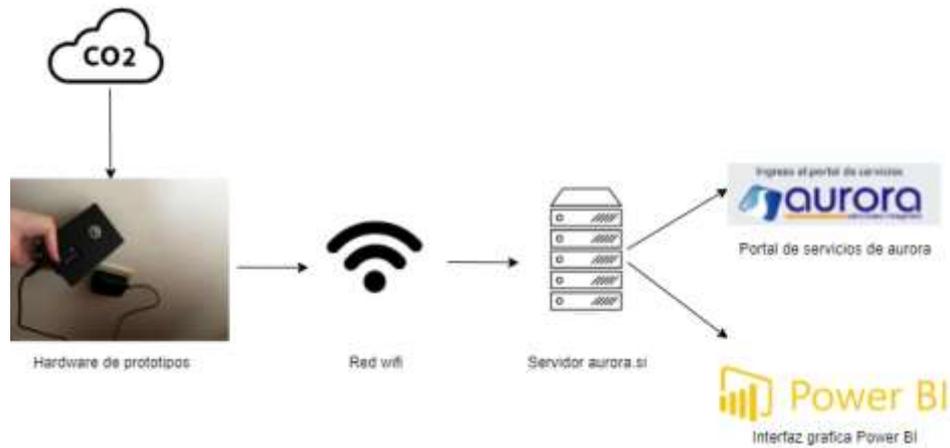


Figura 53. Comunicación de la solución IoT

Una vez que los datos se reflejaban en el Power BI de Aurora, fue enviado a mi persona un documento Power BI con la base de datos previamente asociada. Luego de esto, se diseñaron las pantallas colocando los valores correspondientes en cada gráfico. Para este paso se hizo uso de Power BI Desktop para mayor facilidad. A continuación, se muestran los valores de uno de los gráficos en específico.

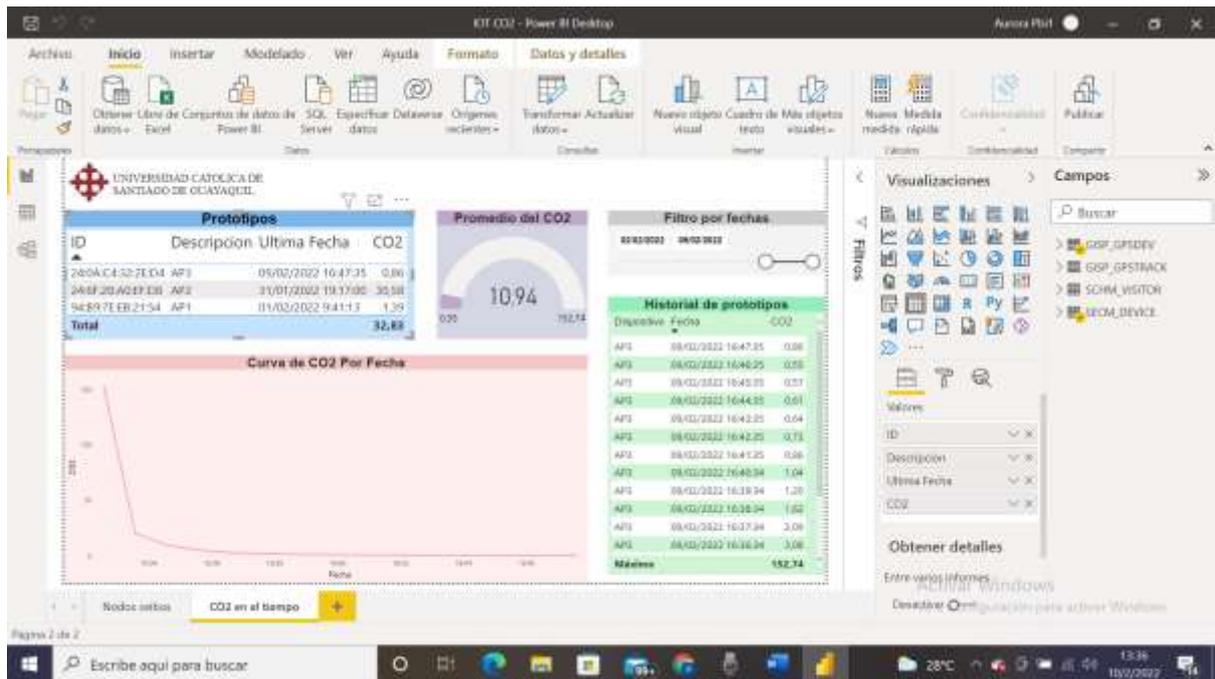


Figura 54. Pantalla de Power BI Desktop

Para finalizar, se mandaron los gráficos diseñados por la autora de este trabajo de titulación para posteriormente ser publicados en el servicio de Power BI en línea y tener la capacidad de observar las actualizaciones. Estas actualizaciones se dan cada hora ya que se cuenta con una suscripción de tipo Pro en Power BI y se permiten un número limitado de actualizaciones al día.

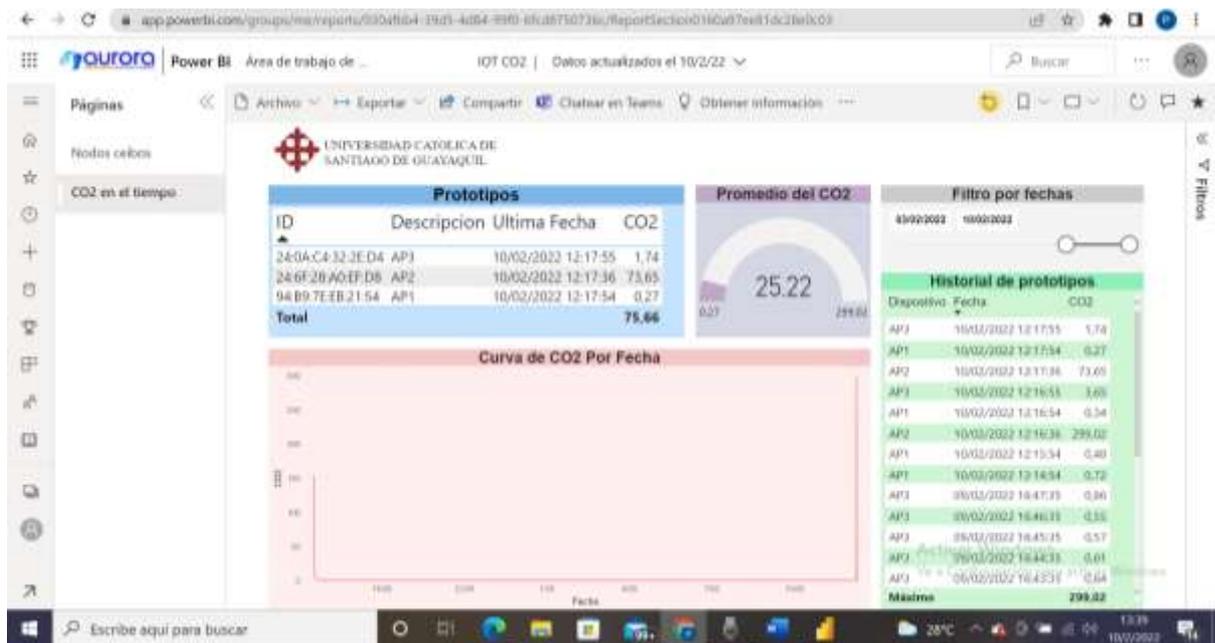


Figura 55. Pantalla del servicio de Power BI Pro

## 4.6. Requerimientos de implementación

### 4.6.1. Hardware

Para la implementación del hardware no fue necesaria la autorización de la parte administrativa de Ceibos Norte, ya que es un proyecto de uso personal y que no tiene como objetivo establecerse como proyecto de la ciudadela ni concientizar a los residentes.

Fue necesario pedirle a los dos residentes permiso para implementar los prototipos en su casa. Adicionalmente encontrar un punto estratégico dentro de la casa que estuviera cerca de la calle, pero a su vez que contara con enchufe eléctrico y red wifi cercana.

## **4.6.2. Software**

### *4.6.2.1. IDE Arduino*

La parte del software de los prototipos fue programada en Arduino IDE. Los requerimientos mínimos para obtener esta plataforma en una PC Windows son:

- Sistema operativo Windows 10 en la versión 14393.0 o más recientes.
- Arquitectura x86.

### *4.6.2.2. Power BI Pro*

En este caso, se va a hacer uso de la versión en línea de Power BI, no hay necesidad de descargar una aplicación. El único requerimiento es tener acceso a internet.

Por otro lado, para este proyecto se necesitaban actualizaciones automáticas de los datos en Power BI, por lo que se optó por una licencia Pro. Este tipo de licencias pueden ser compradas únicamente con cuentas empresariales y deben ser autorizadas por el administrador de la organización.

Para el presente trabajo de titulación se compró la licencia de Power BI por un mes, y se hizo a través de una cuenta prestada de Aurora.

## **4.6.3. Servidor**

Se requirió permiso de la empresa Aurora para hacer uso de sus servicios de red. Adicionalmente fue necesario registrar las direcciones MAC de los prototipos en el portal de servicios de Aurora para que logaran ser visibles.

## 4.7. Materiales

A continuación, se podrán observar las tablas de cotización tanto de hardware como software para la realización y buen funcionamiento de los tres prototipos.

Tabla 1. Materiales para la elaboración del hardware.

<b>Cantidad</b>	<b>Descripción</b>	<b>Precio Unitario</b>	<b>Total</b>
3	Sensor MQ135	\$5.4	\$16.20
3	Arduino Nano	\$9.00	\$27.00
3	Convertidor TTL	\$1.20	\$3.60
3	ESP32	\$12.60	\$37.80
3	Juego de cables jumper	\$2.00	\$6.00
3	Cargadores de 5v	\$5.50	\$16.50
3	Cajas de prototipos	\$8.00	\$24.00
3	Switch	\$0.35	\$1.05
3	Conector	\$0.35	\$1.05
36	Tornillos	\$0.10	\$3.60
3	Placa de circuito PCB	\$10.00	\$30.00
<b>TOTAL</b>			<b>\$166.80</b>

Tabla 2. Materiales para la implementación del software

<b>Cantidad</b>	<b>Descripción</b>	<b>Precio Unitario</b>	<b>Total</b>
1	Licencia mensual Power BI Pro	\$0.00	\$9.99
1	Servicio red Aurora	\$0.00	\$0.00
1	Arduino IDE	\$0.00	\$0.00

<b>TOTAL</b>	<b>\$9.99</b>
--------------	---------------

A continuación, se muestra en la tabla 3 el costo total del trabajo de titulación actual.

Tabla 3. Costo total de materiales para implementación de solución IoT

<b>Materiales</b>	<b>Costo</b>
Elaboración Hardware	\$166.80
Implementación del software	\$9.99
<b>TOTAL</b>	<b>\$176.79</b>

#### 4.8. Estudio Costo/Beneficio

Después de realizar un análisis de la situación actual del Ecuador frente al monitoreo del aire, las repercusiones que tienen los grandes volúmenes de CO2 en el ambiente para nuestra salud y las consecuencias que se pueden ver claramente en el cambio climático debido a los gases de efecto invernadero, podemos concluir que el monitoreo continuo de este gas y de otros puede beneficiarnos de gran manera tanto como personas como sociedad.

Es probable que no en todos los sectores sea de gran utilidad este tipo de proyectos, ya que las fuertes concentraciones se dan en lugares muy concurridos por carros o personas. Por esto considero que los lugares más imprescindibles donde se debería aplicar este tipo de monitoreo deberían ser en carreteras, como la Vía a la Costa, Vía Perimetral, Vía Daule, etc.

Dejando de lado la contaminación ambiental, considero que actualmente el monitoreo del CO2 es de suma importancia, ya que estamos viviendo una pandemia que ha matado millones de personas y una gran forma de prevenir los contagios es manteniendo los niveles de CO2 bajos.

Tomando en cuenta que el monitoreo del aire no solo puede beneficiar al medio ambiente, si no también limitar la transmisión del COVID-19 podemos concluir que es una buena idea invertir en este campo.

Los colegios, universidades y centros comerciales se verían beneficiados al implementar este tipo de sistema en sus instituciones, ya que son opciones económicas que les ayudarían a volver a tener contacto presencial con sus clientes.

#### **4.9. Resultados esperados**

Al poner en práctica la implementación de la propuesta expuesta anteriormente se consiguió llevar a cabo de manera exitosa el desarrollo de la solución IoT de monitoreo del aire en el sector Ceibos Norte.

La representación gráfica en Power BI de los datos extraídos del servidor Aurora, siendo uno de los principales el mapa de la ciudadela con los tres puntos señalados, se ejecutó exitosamente.

Los prototipos funcionaron de manera correcta, enviando los datos al servidor en el tiempo que fueron programados y captando el CO2 a su alrededor con éxito.

## CONCLUSIONES

Mediante todo el proceso del desarrollo de la solución IoT para la ciudadela Ceibos Norte; en el cual se incluye investigación, recolección de requisitos, desarrollo e implantación de los prototipos; podemos concluir que existe una carencia de soluciones IoT en la urbanización, pero también en el Ecuador en general. La propuesta del actual trabajo de titulación se presenta como una alternativa de uso personal o para pequeñas instituciones que están interesadas en mantener un aire limpio que no afecte su salud.

Obedeciendo los objetivos que se plantearon al inicio del actual proyecto, se pudo llegar a las siguientes conclusiones:

- El mayor daño al ambiente y nuestra salud es la desinformación y la falta de acción. Es importante que busquemos maneras de ser menos contaminantes y que el gobierno le de mayor importancia a los efectos nocivos de los gases de efecto invernadero.
- Se logro implementar la solución IoT, recogiendo los datos mediante sensores MQ135 y el hardware que lo acompaña, para finalmente ser enviando hacia el servidor Aurora y mostrado en Microsoft Power BI.
- La herramienta Power BI, muestra los datos en diferentes gráficos de manera correcta y a tiempo real gracias a la suscripción Pro que se compró.
- La cantidad de CO2 en los diferentes puntos seleccionados presenta niveles normales, por lo que podemos concluir que el aire en la ciudadela no perjudica la salud de sus residentes.

## RECOMENDACIONES

Para futuros proyectos de la misma índole que el presente trabajo se recomienda:

- Utilizar directamente la nube de Arduino para cargar los datos a un servidor y ser mostrados posteriormente en su misma plataforma.
- Si se utiliza Power BI, tener contacto directo con el administrador de la compañía con la que se trabaja para lograr comprar la licencia Pro con mayor facilidad.
- Buscar la forma de siempre proteger el hardware del proyecto, ya que puede ser muy sensible a lluvias y suciedad. De preferencia fabricar una caja de prototipado como se hizo en el actual proyecto.

## REFERENCIAS BIBLIOGRÁFICAS

- Mauna Loa Atmospheric Baseline Observatory. (7 de junio de 2021). NOAA. Obtenido de <https://research.noaa.gov/article/ArtMID/587/ArticleID/2764/Coronavirus-response-barely-slows-rising-carbon-dioxide>
- Arduino. (s.f.). Obtenido de Arduino Nano: <https://arduino.cl/arduino-nano/>
- Asociacion de Propietarios de la Urbanizacion Ceibos Norte. (2017). Obtenido de <https://www.ceibosnorte.com/index.php/nosotros/quienes-somos>
- Bhunja, S., & Tehranipoor, M. (2019). *Hardware Security A Hands-On Learning Approach*. Elsevier.
- CDC. (s.f.). *Center for Disease Control and Prevention*. Obtenido de <https://www.cdc.gov/nceh/>
- CEAM. (s.f.). Obtenido de Programa Meteorología Y Contaminación Atmosférica: <http://www.ceam.es/GVAceam/programas/METEO/euphore.htm>
- Cerf, V. (1977). *Specification of Internet Transmission Control Program*. Obtenido de [https://embeddeddiaries.com/esp32-with-tcp-ip-protocol/#:~:text=A%20TCP%20\(transmission%20control%20protocol,delivered%20to%20the%20correct%20destination.](https://embeddeddiaries.com/esp32-with-tcp-ip-protocol/#:~:text=A%20TCP%20(transmission%20control%20protocol,delivered%20to%20the%20correct%20destination.)
- Cortés, M., & Iglesias, M. (2004). *Generalidades sobre Metodología de la Investigación*. Ciudad del Carmen: Universidad Autónoma del Carmen.
- Dannegård, B. (20 de enero de 2021). *Arduino*. Obtenido de Connecting Two Nano Every Boards Through I2C.
- Di Gilio, A., Palmisani, J., Pulimeno, M., Cerino, F., Cacace, M., Miani, A., & de Gennaro, G. (3 de julio de 2021). *CO2 concentration monitoring inside educational buildings as a strategic tool to reduce the risk of Sars-CoV-2 airborne transmission*. Milan: Elsevier. Obtenido de <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8253691/>
- El Universo*. (27 de julio de 2021). Obtenido de Medir la calidad del aire y el impacto del ruido en 23 puntos de Guayaquil costará \$ 159.360 al año.
- El Universo*. (27 de julio de 2021). Obtenido de El Municipio medirá la calidad del aire y el impacto del ruido en 23 puntos de Guayaquil, definidos en consultoría: <https://www.pressreader.com/ecuador/el-universo/20210727/281728387550481>

*Electrónica Caribe*. (s.f.). Obtenido de Cable Jumper Para Arduino:  
<https://electronicacaribe.com/product/cable-jumper-para-arduino/>

EPA. (s.f.). *Environmental Protection Agency*. Obtenido de Greenhouse Gas Emissions from a Typical Passenger Vehicle: <https://www.epa.gov/greenvehicles/greenhouse-gas-emissions-typical-passenger-vehicle#:~:text=typical%20passenger%20vehicle%3F-A%20typical%20passenger%20vehicle%20emits%20about%204.6%20metric%20tons%20of,8%2C887%20grams%20of%20CO2>.

Fundación Aequae. (s.f.). *Fundación Aequae*. Obtenido de ¿Qué es la contaminación ambiental?:  
<https://www.fundacionaequae.org/causas-contaminacion-ambiental/>

Giraldo, S. A. (s.f.). *Control Automático Educación*. Obtenido de ADC Arduino – Entradas Analógicas: <https://controlautomaticoeducacion.com/arduino/entradas-analogicas-adc/#:~:text=Que%20es%20adc%20en%20arduino,proyecto%20de%20automatizaci%C3%B3n%20con%20Arduino>.

Hübscmann, I. (28 de agosto de 2020). *Nabto*. Obtenido de ESP32 for IoT: A Complete Guide:  
<https://www.nabto.com/guide-to-iot-esp-32/>

*iberotecno*. (21 de Febrero de 2021). Obtenido de El sensor de CO2 MQ-135 analizado al detalle: <https://iberotecno.com/tienda/blog/electronica/el-sensor-de-co2-mq-135-analizado-al-detalle>

IBM Cloud Education. (8 de mayo de 2019). *IBM*. Obtenido de What is Java?:  
<https://www.ibm.com/cloud/learn/java-explained>

Java. (s.f.). *Java*. Obtenido de [https://www.java.com/en/download/help/whatis\\_java.html](https://www.java.com/en/download/help/whatis_java.html)

Keim, R. (10 de abril de 2020). *All about circuits*. Obtenido de What Is a Printed Circuit Board (PCB)?: <https://www.allaboutcircuits.com/technical-articles/what-is-a-printed-circuit-board-pcb/>

Leahy, P. (20 de febrero de 2019). *thoughtco*. Obtenido de Developing a Java GUI:  
<https://www.thoughtco.com/gui-2034108>

Lee, E. (s.f.). *Techwalla*. Obtenido de How Does an AC Adaptor Work?:  
<https://www.techwalla.com/articles/how-does-an-ac-adaptor-work>

Lewis, S. (s.f.). *TechTarget*. Obtenido de Gateway:  
<https://internetofthingsagenda.techtarget.com/definition/gateway>

- Madhu. (10 de abril de 2020). *Codrey Electronics*. Obtenido de Types of Switches in Electronics: <https://www.codrey.com/electronics/types-of-switches-in-electronics/>
- MAE. (s.f.). *Gobierno del encuentro*. Obtenido de Programa Calidad del Aire Fase III: <https://www.ambiente.gob.ec/programa-calidad-del-aire-fase-iii/>
- MAE. (s.f.). *Ministerio del Ambiente, Agua y Transición Ecológica*. Obtenido de MAE trabaja en programas de mitigación y adaptación para reducir emisiones de Co2 en Ecuador: <https://www.ambiente.gob.ec/mae-trabaja-en-programas-de-mitigacion-y-adaptacion-para-reducir-emisiones-de-co2-en-ecuador/>
- McCracken, D., & Jackson, M. (1982). *Life cycle concept considered harmful*. New York: Association for Computing Machinery.
- Microsoft. (s.f.). Obtenido de What is business intelligence?: <https://powerbi.microsoft.com/en-us/what-is-business-intelligence/>
- Microsoft. (21 de septiembre de 2021). Obtenido de What is Power BI Report Server?: <https://docs.microsoft.com/en-us/power-bi/report-server/get-started>
- Microsoft. (23 de septiembre de 2021). Obtenido de Power BI Report Builder: <https://docs.microsoft.com/en-us/power-bi/paginated-reports/report-builder-power-bi>
- Microsoft. (21 de septiembre de 2021). *Microsoft*. Obtenido de What is Power BI?: <https://docs.microsoft.com/en-us/power-bi/fundamentals/power-bi-overview>
- Motisan, R. (1 de octubre de 2016). *uRADMonitor*. Obtenido de Air Quality Monitoring over LoraWAN: <https://www.uradmonitor.com/air-quality-monitoring-lorawan/>
- Naylamp. (s.f.). Obtenido de Conversor De Nivel Lógico 4CH 5V/3.3V: <https://naylampmechatronics.com/conversores-ttl/67-conversor-de-nivel-logico-4ch-5v3v.html>
- Oracle. (s.f.). *Oracle*. Obtenido de What Can Java Technology Do?: <https://docs.oracle.com/javase/tutorial/getStarted/intro/cando.html>
- Pachauri, R., & Reisinger, A. (2008). *Climate Change 2007*. Switzerland: Intergovernmental Panel on Climate Change. Obtenido de AR4 Climate Change 2007: Synthesis Report: <https://www.ipcc.ch/report/ar4/syr/>
- Paz, H. G. (2000). *Investigación Científica*. Quito: Ediciones Abya-Yala.

Real Academia Española. (2021). *RAE*. Obtenido de metodología: <https://dle.rae.es/metodolog%C3%ADa>

*Romanian Startups*. (s.f.). Obtenido de Environmental Pollution Monitoring: <https://www.romanianstartups.com/startup/uradmonitor/>

Santos, E. (03 de mayo de 2021). *Medidor de CO2*. Obtenido de Medidor de CO2 con Arduino: <https://medidordeco2.info/medidor-de-co2-arduino/>

Smith, D. (6 de septiembre de 2019). *kaiterra*. Obtenido de CO2 Sensors: Which Type Should You Be Looking For?: <https://learn.kaiterra.com/en/air-academy/carbon-dioxide-sensors>

The Power BI Team. (10 de julio de 2015). *Microsoft*. Obtenido de Announcing Power BI general availability coming July 24th: <https://powerbi.microsoft.com/en-us/blog/announcing-power-bi-general-availability-coming-july-24th/>

Westreicher, G. (12 de septiembre de 2020). *Economipedia*. Obtenido de Diferencia entre método y metodología.: <https://economipedia.com/definiciones/diferencia-entre-metodo-y-metodologia.html>

*Wisconsin Department of Health Services*. (3 de junio de 2021). Obtenido de <https://www.dhs.wisconsin.gov/chemical/carbondioxide.htm#:~:text=400%20ppm%3A%20average%20outdoor%20air,stagnant%2C%20stale%2C%20stuffy%20air>.

World Health Organization. (2021). Air pollution. En *Compendium of WHO and other UN guidance on health and environment* (pág. 24). Switzerland.

World Health Organization. (s.f.). *WHO*. Obtenido de <https://www.who.int/data/gho/>

Wright, N. (s.f.). *Nigel Frank*. Obtenido de Everything you ever wanted to know about Microsoft Power BI: <https://www.nigelfrank.com/blog/everything-you-ever-wanted-to-know-about-microsoft-power-bi/>

## **APENDICES**

### **Apéndice A. Formato de ENTREVISTA: El CO2 y sus efectos nocivos en la salud y el ambiente**

1. ¿Qué efectos negativos para la salud puede tener el exceso de este gas?
2. ¿Qué podemos hacer las personas para disminuir la emisión del CO2?
3. ¿Qué puede hacer el gobierno para disminuir las emisiones del CO2?
4. ¿Cree que en Ecuador es importante tomar acciones para frenar la contaminación ambiental y el cambio climático producido por este gas de efecto invernadero?

## **Apéndice B. Formato de ENTREVISTA: El internet de las cosas y la contaminación ambiental**

1. ¿Cree que el internet de las cosas pueda llegar a cambiar nuestras vidas drásticamente?
2. ¿Que otro tipo de soluciones IoT existen para la conservación ambiental?
3. ¿Considera que el monitoreo del aire mediante soluciones IoT es importante hoy en día?
4. ¿Considera que el IoT tiene una buena relación entre costo/beneficio?

# ANEXOS

```
// CÓDIGO PARA ARDUINO
#include <MQ135Sensor.h>
#include <SoftwareSerial.h>
#include <Wire.h>

String message;
bool B = false;

String header = "120*";

#define placa "Arduino Uno"
#define Voltage_Resolution 5
#define pin A0 //Analog input 0 of your arduino
#define type "MQ-135" //MQ135
#define ADC_BIT_RESOLUTION 10 // For arduino UNO/MEGA/NANO
#define RatioMQ135CleanAir 3.6//B0 // B0 = 3.6 ppm

double CO2 = 0;

MQ135Sensor MQ135(placa, Voltage_Resolution, ADC_BIT_RESOLUTION, pin, type);

void MQ135_setup() {
  //set math model to calculate the PPM concentration and the value of constants
  MQ135.setRegressionMethod(1); // PPM = a*ratio^b
  MQ135.setA(110.47); MQ135.setB(-2.962); // Configure the equation values to get CO2 concentration
}
```

Figura 56. Parte 1 del código del programa de Arduino Nano

```
MQ135.init();
//
//!! the B1 value is different from B0 please change your B1 value with the following method:
MQ135.setB1(B1);
//
//*****MQ135CleanAir *****
// Explanation:
// In this routine the sensor will measure the resistance of the sensor supposing before was pre-heated
// and now is in clean air (Calibration conditions), and it will setup B0 value.
// We recommend execute this routine only on setup or on the laboratory and save on the memory of your arduino
// This routine not need to execute at every context, you can read your B0 if you know the value
// Acknowledgments: https://www.gammasystems.com/blog/understanding-a-gas-sensor
Serial.println("Calibrating please wait.");
float caldB0 = 0;
const int i = 1; //10; i++;
{
  MQ135.update(); // Update data, the arduino will be read the voltage on the analog pin
  caldB0 += MQ135.calibrate(RatioMQ135CleanAir);
  Serial.print(" ");
}
MQ135.setB0(caldB0/10);
Serial.println(" done.");

if(!digitalRead(B1)) {Serial.println("Warning: Connection issue founded, B0 is null (Open circuit detected) please check your wiring and supply"); while(1);}
if(caldB0 == 0){Serial.println("Warning: Connection issue founded, B0 is zero (Analog pin with short circuit to ground) please check your wiring and supply"); while(1);}
```

Figura 57. Parte 2 del código del programa de Arduino Nano

```

MQ135.serialDebug(Fatal);
}

void Sensor_MQ135 () {
  MQ135.update(); // Update data, the arduino will be read the voltage on the analog pin
  // MQ135.setA(110,47); MQ135.setB(2.862);
  CO2 = MQ135.readSensor();
  Serial.print("CO2: ");
  Serial.println(CO2);
  //delay(1000);
}

void setup() {
  Serial.begin(9600);
  //usb.begin(4800); // 9600/4800/2400
  //pinMode(FIN, OUTPUT);

  Wire.begin(23); //23 // unirse al bus i2c con la direccion #23
  Wire.setClock(400000);
  Wire.onRequest(eventoSolicitud); // registrar evento de solicitud de datos
  Wire.onReceive(eventoRecepcion); // registrar evento de recepcion de datos

  MQ135_setup();
}

```

Figura 58. Parte 3 del código del programa de Arduino Nano

```

void loop() {
  delay(500);
}

void eventoRecepcion() {
  ////////////////////////////////////////////////////////////////////
  // if( Wire.read() == 'R' ) {
  //   S = true;
  //   message = String( millis() ) + "\n";
  // }
  ////////////////////////////////////////////////////////////////////
  if( Wire.read() == 'R' ) {
    S = true;

    // poner la funcion de los sensores
    Sensor_MQ135();
    Mensaje();
    //message = String( millis() ) + "\n";
  }
}

// function that executed whenever data is requested by master
void eventoSolicitud() {
  if( S == true ) {
    Wire.write(message.length());
  }
}

```

Figura 59. Parte 4 del código del programa de Arduino Nano

```

////////////////////////////////////
if( Wire.read() == '0' ){
  E = true;

  // para la función de los sensores
  sensor_MQ135();
  Mensaje();
  //mensaje = String( milis() ) + '\n';
}
}

// función que ejecuta siempre que se ejecuta la función de master
void sensorMilitaridad() {}

if( E == true ){
  Wire.write( mensaje, sizeof( mensaje ) );
  E = false;
}
else{
  Wire.println( mensaje );
}
}

void Mensaje() {
  //mensaje = String( sensor1 + "," + String( sensor2 + "," + String( sensor3 );
  mensaje = header + "," + String( 000 );
}
}

```

Figura 60. Parte 5 del código del programa de Arduino Nano

```

#include <Wire.h>
#include <WiFi.h>

//PINES PARA I2C por defecto
#define I2C_SDA 21
#define I2C_SCL 22

// pines para I2C
#define I2C_SDA2 27
#define I2C_SCL2 28

TwoWire I2C = TwoWire(1);
//-----
String Nombre_Equipo = "API"; // Nombre de equipo = prototipo
const char* ssid = "D-Link_DLR-615"; // Nombre de la red wifi
const char* password = "911791101463"; // Contraseña de la red wifi
//-----
const uint8_t port = 3022;
const char* host = "aurora1.net";
WiFiClient client;
String IoTdata;
String startbit = "SIOT";

static uint32_t lastMillis = 0;

```

Figura 61. Parte 1 del código del programa de ESP32

```

static uint32_t lastMillis = 0;
//-----
int tiempo = 1; // configuración del tiempo de envío de datos en minutos
//-----
//OLED_MQ135(11, 13);
int i = 0;

extern uint8_t smallFont[];
//extern uint8_t TinyFont[];

//byte pin[] = {1};
//byte estado = 0;
int retardo = 10000;
String cadena;
//int ValorSensor = 0;
String header = "IoT";
float TCE;

void IoTMessage( String strtoken ) {
  strtoken.trim();
  if ( strtoken.startsWith( header ) ) {
    Serial.println( "Esen detectado!" + strtoken );
    //-----convertir de string a char-----
    char buff[ strtoken.length() + 1 ];
    char *str = buff;

```

Figura 62. Parte 2 del código del programa de ESP32

```

int index = 0;
strtok_r(buff, "\n", &ptr); // takes a list of delimiters
while (ptr != NULL)
{
  if (index == 0) // el primer es la cabecera
  {
    Serial.println(ptr);
    //SPC.Add(ptr);
    string value = string(ptr);
    if (index == 1)
      cmd = value.trim();
  }
  index++;
  ptr = strtok(NULL, "\n"); // takes a list of delimiters
}
Serial.print("CMD: ");
Serial.println(cmd);
Serial.println("sendMessage Finalizado");
}
else {
  Serial.println("Cabecera invalida");
}
}

void WiFiSetup() {
  Serial.println("Conectando a WiFi...");
}

```

Figura 63. Parte 3 del código del programa de ESP32

```

Serial.println(ssid);
Serial.println(password);
Serial.println(password);

WiFi.begin(ssid, password);
delay(1000);
int contador = 0;
while (WiFi.status() != WL_CONNECTED && contador < 10) {
  Serial.println("Esperando conexión WiFi : " + String(contador) + " Status: " + String(WiFi.status()));
  delay(1000);
  contador++;
}
//
// para el caso
WL_NO_SHIELD = 0, // for compatibility with WiFi shield library
WL_IDLE_STATUS = 0,
WL_NO_SSID_AVAIL = 1,
WL_SCAN_COMPLETED = 2,
WL_CONNECTED = 3,
WL_CONNECT_FAILED = 4,
WL_CONNECTION_LOST = 5,
WL_DISCONNECTED = 6
} WL_STATUS_t;
//
contador = 0;
if (WiFi.status() == WL_CONNECTED) {
  Serial.println("WiFi connected with IP: ");
}

```

Figura 64. Parte 4 del código del programa de ESP32

```

Serial.println(WiFi.localIP());
Serial.println("ESP32 Board MAC Address: ");
Serial.println(WiFi.macAddress());
connectToServer();
} else if (WiFi.status() == WL_IDLE_STATUS) {
  Serial.println("WiFi No disponible !!!!");
}
else {
  Serial.println("WiFi Status:");
  Serial.println(WiFi.status());
  Serial.println("WiFi No disponible !!!!");
}

void connectToServer() {
  if (WiFi.status() == WL_NO_SHIELD) {
    Serial.println("WiFi shield not present");
    Serial.println("Connection impossible");
    // don't continue:
    return;
  }
  if (WiFi.status() == WL_CONNECTED) {
    Serial.println("Connecting to server");
    Serial.println(host);
    Serial.println("");
    Serial.println(port);
    if (!client.connect(host, port)) {
      Serial.println("Connection fail !!!!!");
    }
  }
}

```

Figura 65. Parte 5 del código del programa de ESP32

```

        Serial.println("Connection successful!");
    }
}
else {
    Serial.print("WiFi is not available (11) ");
    Serial.print("Reintento conectar ");
    WiFiSetup();
}
}

void muestraIP() {
    IPAddress = IPAddress("WiFi.localName()","192.168.1.100","192.168.1.100");
    Serial.println(IP);
    Serial.println("Direccion por ip: ");
    Serial.println(IP);
    Serial.println("-----");
    Serial.println("-----");
    Serial.println(IP); //Servir la pagina
}

void procesaComando() {
    if (WiFi.status() == WL_CONNECTED) {
        if (isSerialConnected()) {
            Serial.println("Comando disponible");
        }
    }
}

```

Figura 66. Parte 6 del código del programa de ESP32

```

        Serial.println("Se va a servir pagina ip: ");
        if (isSerialConnected()) {
            muestraIP();
        }
    }
    else {
        Serial.println("No conectado al puerto TCP");
    }
}
else {
    Serial.println("Conexion WiFi no disponible");
}
}

void TCP_request() {
    // Serial.println("Mensaje recibido");
    //Solo se recibe conectado ya TCP
    TCP_requestFromClient(); //Mensaje recibido = byte
    TCP_write("OK");
    TCP_writeFromClient();
    TCP_requestFromClient(); // Solicitar a byte del esclavo A12
    byte req = TCP_read();
    TCP_requestFromClient(); // Solicitar 'lan' byte del esclavo A12
    while (TCP_read() < 0) { // slave may send back that requested
    }
}

```

Figura 67. Parte 7 del código del programa de ESP32

```

        cadena.concat(String(TCP_read()));
        //char c = TCP_read(); // devuelve a byte or character
        //Serial.println(c); // plus, the character
        Serial.println(cadena);
        mensaje(cadena);
        cadena="";
    }
}

void setup() {
    //Wire.begin(10, 10, 10);
    Serial.begin(115200);
    //TCP.begin(TCP_IP, TCP_PORT, 8080);
    TCP.begin(TCP_IP1, TCP_PORT1, 4080);
    // /!#! setup()
    WiFiSetup();
}

void loop() {
    TCP_request();
    delay(100);
    if (millis() - lastMillis > 1000) {lastMillis = millis();
        Serial.println("Reinicio Time");
        procesaComando();
    }
}

```

Figura 68. Parte 8 del código del programa de ESP32

## DECLARACIÓN Y AUTORIZACIÓN

Yo, **Burbano Delgado, María Paula** con C.C: # **0926065863** autor/a del trabajo de integración curricular: **Interfaz gráfica de monitoreo de la calidad del aire en un sector residencial de Guayaquil** previo a la obtención del título de **Ingeniero en Ciencias de la Computación** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, **02 de marzo del 2022**



f. \_\_\_\_\_

Nombre: **Burbano Delgado, María Paula**

C.C: **0926065863**



## REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

### FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

<b>TEMA Y SUBTEMA:</b>	Interfaz gráfica de monitoreo de la calidad del aire en un sector residencial de Guayaquil		
<b>AUTOR(ES)</b>	Burbano Delgado, María Paula		
<b>REVISOR(ES)/TUTOR(ES)</b>	Ing. Céleri Mujica, Colón Mario , Mgs		
<b>INSTITUCIÓN:</b>	Universidad Católica de Santiago de Guayaquil		
<b>FACULTAD:</b>	Ingeniería		
<b>CARRERA:</b>	Ingeniería en Ciencias de la Computación		
<b>TÍTULO OBTENIDO:</b>	Ingeniero en Ciencias de la Computación		
<b>FECHA DE PUBLICACIÓN:</b>	02 de marzo de 2022	<b>No. DE PÁGINAS:</b>	86
<b>ÁREAS TEMÁTICAS:</b>	Ingeniería en Computación		
<b>PALABRAS CLAVES/KEYWORDS:</b>	Arduino Nano, placa ESP32, sensor MQ135, IOT, Aurora, Power BI.		
<b>RESUMEN/ABSTRACT:</b>	<p>El papel que desarrollan los humanos en la tierra es la principal razón de que la contaminación cada vez sea más preocupante. Dado que esta situación no ha mejorado en las últimas décadas, se ha vuelto imprescindible el control y monitoreo de la calidad del aire en el ambiente. Los monitoreos del aire se realizan para asegurar la salud de los ciudadanos al igual que la conservación de la flora y fauna que nos rodea, también es una manera de evitar el deterioro urbanístico de las ciudades. El objetivo de este proyecto es monitorear el aire de la ciudadela Ceibos Norte, mediante una solución IoT utilizando las placas Arduino Nano y ESP32. El servidor que se escogió para enviar la información recolectada es Aurora, y la herramienta para la visualización de los resultados de manera grafica fue Power BI. Para la realización del actual proyecto se utilizó una metodología mixta, dándole uso a entrevistas y sacando conclusiones de los valores que los prototipos IoT arrojan. Mediante las entrevistas se pudo concluir que se necesita concientizar más a la población ecuatoriana sobre la importancia de la conservación ambiental y como los avances tecnológicos pueden ayudarnos a reducir el impacto negativo que tenemos sobre el medio ambiente. Además, se utilizó la metodología de diseño rápido para el prototipado donde se definió la arquitectura del mismo, lo que fue de ayuda para realizar los diferentes prototipos de manera correcta. El conjunto de las metodologías dio resultados positivos y satisfactorios para la solución IoT del presente trabajo de titulación.</p>		
<b>ADJUNTO PDF:</b>	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
<b>CONTACTO CON AUTOR/ES:</b>	<b>Teléfono:</b> 0979408925	<b>E-mail:</b> mariapaulaburbano2000@hotmail.com	
<b>CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):</b>	<b>Nombre:</b> Ing. Toala Quimí, Edison José		
	<b>Teléfono:</b> +593-042 20 27 63 / 593-9-90976776		
	<b>E-mail:</b> edison.toala@cu.ucsg.edu.ec		
<b>SECCIÓN PARA USO DE BIBLIOTECA</b>			
<b>Nº. DE REGISTRO (en base a datos):</b>			
<b>Nº. DE CLASIFICACIÓN:</b>			
<b>DIRECCIÓN URL (tesis en la web):</b>			