

**UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ**

**Colegio de Ciencias e Ingenierías**

**Sistema de monitoreo de contaminación del aire con redes  
de sensores inalámbricos en tiempo real**

**Artículo Académico**

**Diego Rodrigo Ortiz Espinosa**

**Ingeniería Electrónica**

Trabajo de titulación presentado como requisito  
para la obtención del título de  
Ingeniero Electrónico

Quito, 17 de mayo de 2018

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ  
COLEGIO DE CIENCIAS E INGENIERÍAS

**HOJA DE CALIFICACIÓN  
DE TRABAJO DE TITULACIÓN**

**Sistema de monitoreo de contaminación del aire con redes de sensores  
inalámbricos en tiempo real**

**Diego Rodrigo Ortiz Espinosa**

Calificación:

Nombre del profesor, Título académico

Diego Benítez, Ph.D.

Firma del profesor

---

Quito, 17 de mayo de 2018

## Derechos de Autor

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Firma del estudiante:

---

Nombres y apellidos:

Diego Rodrigo Ortiz Espinosa

Código:

00116790

Cédula de Identidad:

1723300255

Lugar y fecha:

Quito, 17 de mayo de 2018

## RESUMEN

Este proyecto consiste en la creación de un sistema de monitoreo ambiental de bajo costo con redes de sensores inalámbricos visualizados en tiempo real por medio de una aplicación web. Los nodos de medición toman datos de CO<sub>2</sub>, partículas de polvo (PM<sub>2.5</sub>) y temperatura presente en el ambiente. El sistema funciona de tal manera que los valores medidos por los sensores son codificados y enviados vía Wireless hacia el Gateway (Raspberry pi) por medio de los módulos Xbee pro S1. Los datos son decodificados y almacenados en la base de datos. Posteriormente, la aplicación web realiza una conexión PHP con la base de datos, los datos son graficados en tiempo real, con el uso de la librería de JavaScript de HighChart y con métodos AJAX y notación JSON. Par comprobar el funcionamiento del sistema, se tomaron mediciones de las concentraciones de CO<sub>2</sub>, PM<sub>2.5</sub> y temperatura dentro de un parqueadero subterráneo y al aire libre.

Palabras clave: Calidad, aire, monitoreo, CO<sub>2</sub>, PM<sub>2.5</sub>, Xbee, json, tiempo real.

## ABSTRACT

This project consists in the creation of a low cost environmental monitoring system with wireless sensor networks visualized in real time by a web application. The measuring nodes take data of CO<sub>2</sub>, dust particles (PM<sub>2.5</sub>) and temperature present in the environment. The system works in such way that the values measured by the sensors are encoded and sent via Wireless to the Gateway (Raspberry pi) by the Xbee Pro S1 modules. The data is decoded and stored in the database. Afterwards, the web application makes a PHP connection with the database and the data is plotted in real time, with the use of the HighChart JavaScript library with AJAX methods and JSON notation. To test the system, measurements concentrations of CO<sub>2</sub>, PM<sub>2.5</sub> and temperature were taken inside and outside an underground parking lot.

*Key words:* air quality, monitoring system, CO<sub>2</sub>, PM<sub>2.5</sub>, Xbee, json, real time.

## TABLA DE CONTENIDO

<b>Introducción .....</b>	<b>7</b>
<b>Metodología .....</b>	<b>8</b>
<b>Resultados .....</b>	<b>11</b>
<b>Conclusiones.....</b>	<b>12</b>
<b>Referencias bibliográficas .....</b>	<b>12</b>

# Sistema de Monitoreo de Contaminación del Aire con Redes de Sensores Inalámbricos en Tiempo Real

Diego Ortiz, Diego Benítez  
 Universidad San Francisco de Quito USFQ,  
 Colegio de Ciencias e Ingenierías "El Politécnico",  
 Campus Cumbayá, Casilla Postal 17-1200-841, Quito, Ecuador  
 diego.ortiz@estud.usfq.edu.ec, dbenitez@usfq.edu.ec

**Resumen**—Este proyecto consiste en la creación de un sistema de monitoreo ambiental de bajo costo con redes de sensores inalámbricos visualizados en tiempo real por medio de una aplicación web. Los nodos de medición toman datos de CO<sub>2</sub>, partículas de polvo (PM<sub>2.5</sub>) y temperatura presente en el ambiente. El sistema funciona de tal manera que los valores medidos por los sensores son codificados y enviados vía Wireless hacia el Gateway (Raspberry pi) por medio de los módulos Xbee pro S1. Los datos son decodificados y almacenados en la base de datos. Posteriormente, la aplicación web realiza una conexión PHP con la base de datos, los datos son graficados en tiempo real, con el uso de la librería de JavaScript de HighChart y con métodos AJAX y notación JSON. Par comprobar el funcionamiento del sistema, se tomaron mediciones de las concentraciones de CO<sub>2</sub>, PM<sub>2.5</sub> y temperatura dentro de un parqueadero subterráneo y al aire libre.

**Index Terms**—Calidad, aire, monitoreo, CO<sub>2</sub>, PM<sub>2.5</sub>, Xbee, json, tiempo real.

## I. INTRODUCCIÓN

La contaminación ambiental es uno de los mayores problemas mundiales que afecta la salud humana. Aproximadamente 7 millones de personas mueren por año debido a enfermedades relacionadas con la contaminación del aire [1]. Con el crecimiento de la población y de las fuentes de emisión de contaminantes, es un desafío cada vez mayor el control de la calidad del aire [2]. En este sentido, la información que puede proveer en tiempo real un sistema de monitoreo de calidad del aire es de gran utilidad para: poder determinar lugares en donde se tenga una alta exposición a contaminantes, analizar el peligro que pueden correr las personas que estén en exposición constante debido a los contaminantes ambientales presentes, evitar graves consecuencias a los ecosistemas y seres vivos del planeta y promover la conciencia ambiental.

Las redes de sensores inalámbricos (WSN) han sido de gran utilidad para lograr crear centros de monitoreo ambientales de bajo costo [3]. Las ventajas de utilizar estas redes de sensores son: que se puede implementar un mayor número de nodos de medición para aumentar el área de monitoreo, tienen una instalación sencilla, son de bajo mantenimiento y son apropiadas para recolectar información. Sin embargo, una de sus mayores desventajas es que dependen de una fuente de poder limitada, debido a que estos nodos se pueden encontrar en un lugar remoto con difícil acceso a fuentes de poder tradicionales.

La actividad contaminante que tiene la mayor cantidad de emisiones contaminantes como óxidos, monóxido y dióxido

de carbono es el transporte. Esta actividad afecta directamente a la calidad del aire de una ciudad. Las emisiones de gases contaminantes que produce el transporte diario de las personas tienen un gran impacto en las capas de la atmósfera más bajas, por lo que afecta directamente a la salud de las personas [6]. En los últimos años la cantidad de vehículos ha aumentado significativamente en zonas urbanas y rurales, por lo que es necesario medir la cantidad de gases contaminantes que estos emiten.

En la literatura, se han propuesto modelos de monitoreo ambiental, en donde se utilizaron sensores de CO<sub>2</sub> y NO<sub>2</sub> con salidas analógicas para medir las concentraciones de gases que se emiten cerca de una motocicleta, y también en un lugar cerrado lleno de personas [4]. También se han propuesto modelos de monitoreo en tiempo real en donde se presenta una solución acoplada sobre una unidad móvil, que puede medir los niveles de gases del ambiente en tiempo real, cuyo nodo consiste en un sensor inalámbrico que está asociado a un teléfono inteligente para actuar como interfaz [5].

Existen varios centros de medición de calidad de aire en donde se utilizan equipos de medición de gases atmosféricos muy sofisticados y de alto costo, sin embargo, en este trabajo se propone una alternativa en donde se puede tener un mayor número de nodos de medición con un presupuesto mucho más bajo y de fácil implementación. Al tratarse de sensores de bajo costo, las concentraciones de los gases a medir deben estar dentro del rango de sensibilidad de los sensores. Las mediciones ambientales se realizarán en lugares donde se cumplan con estos rangos de sensibilidad para garantizar la fiabilidad de los datos y el correcto funcionamiento de los sensores.

Este proyecto consiste en la creación de un sistema de monitoreo de gases ambientales diseñado para la recolección, análisis y almacenamiento de mediciones de gases perjudiciales para la salud humana, tales como: partículas de polvo PM<sub>2.5</sub> y dióxido de carbono CO<sub>2</sub>. Los datos medidos por los sensores serán enviados vía Wireless por medio de los módulos de comunicación Xbee desde la placa de adquisición de datos (Arduino) hacia el puerto de salida (Raspberry Pi), los datos serán decodificados y almacenados en una base de datos. El sistema cuenta con un servidor web que realiza una conexión PHP para hacer consultas de registros a la base de datos. Los datos son transformados a formato JSON y graficados en tiempo real con métodos AJAX y la librería JavaScript de

Highchart.

## II. METODOLOGÍA

El sistema de monitoreo de contaminación ambiental está compuesto por nodos de medición que contienen 2 sensores de gases ambientales, un Arduino nano, un Raspberry Pi (Gateway), y una PC que sirve como servidor web para la aplicación en línea. En la Fig. 1 se muestra la topología del sistema de monitoreo implementado. Una ventaja de utilizar esta topología para mediciones ambientales es que los nodos de sensores pueden estar en lugares remotos con altos índices de concentraciones, en donde no haya conexión a Internet y fuentes de alimentación, ya que los datos recogidos por la tarjeta electrónica son transmitidos vía Wireless por los módulos Xbee teniendo un alcance de hasta 500m hacia el Gateway en lugares abiertos. Además, otra gran ventaja de esta topología es que, si se interrumpe la conexión a Internet o la conexión a la red local, los datos recibidos no se pierden ya que son escritos localmente en la base de datos independientemente de la conexión de red.

Los datos medidos por los sensores son codificados en una trama y son enviados vía Wireless hacia el Gateway (Raspberry Pi), en donde se decodifican y son guardados en la base de datos. Posteriormente, una PC que contiene el servidor de aplicaciones Apache, conectada a la misma red LAN del Raspberry Pi, por medio de una conexión PHP lee los registros disponibles en la base de datos. Estos datos son graficados en tiempo real, con el uso de la librería de JavaScript de HighChart y con métodos AJAX y notación JSON. Por último, con ayuda del toolkit de NO-IP.com DUC 4.1.1 [10] se logra la configuración deseada para la publicación en internet de la página web que muestra los datos en tiempo real. Los detalles de este procedimiento se describen a continuación.

### II-A. Configuración comunicación Wireless

Para la comunicación Wireless entre el nodo de adquisición de datos y el Raspberry pi (Gateway) se utilizó dos módulos Xbee Pro S1 802.15.4 [11], los cuales están configurados para tener una comunicación punto a punto. Para esto es necesario que un módulo Xbee esté configurado como coordinador y otro como Punto de salida (End Point). Con el programa XCTU se procede a configurar los módulos en donde los parámetros primordiales a definir para tener comunicación entre los dos dispositivos son: Personal Area Network ID (PAN ID), Destination High (DH) y Low Address (DL). Se requiere que el PAN ID sea el mismo en ambos módulos para que tengan comunicación dentro de una misma red inalámbrica. Como se trata de un enlace punto a punto es necesario definir la dirección de destino en cada módulo. En este caso, en cada dispositivo se ingresa la dirección (DL) del módulo con el cual se va a comunicar y viceversa. La dirección de destino alta (DH) es la misma para dispositivos de la misma familia comercial, en este caso el modelo Xbee Pro S1.

Los parámetros utilizados en la comunicación serial de los módulos Xbee, Arduino Nano y Raspberry Pi fueron los

siguientes: 9600 Baud Rate, sin paridad, un bit de parada y ocho bits de datos.

### II-B. Conexión y configuración de los sensores

Para la adquisición de datos medidos por los sensores, se utilizó un Arduino Nano, que contiene el microprocesador ATmega328P [12]. Este modelo tiene 6 entradas análogas de 5V, y cuenta con un conversor análogo digital de 10 bits de resolución. Los módulos sensores cuentan con salidas análogas que fueron conectadas a las entradas analógicas del Arduino para su lectura. Además, fueron polarizados con 5 voltios que suministra el Arduino Nano.

El sensor utilizado para la medición de CO2 fue el MQ135 y para las partículas de polvo PM2.5 el sensor Sharp GP2Y10. En la Tabla I se muestra los rangos de medición de estos sensores. El sensor de CO2 tiene que ser precalentado durante un periodo de 12 horas para su correcto funcionamiento. El sensor óptico de polvo Sharp GP2Y10 cuenta con un diodo infrarrojo y con un fototransistor alineados para tomar mediciones de polvo. Para lograr buenas mediciones con este sensor, es necesario controlar el diodo infrarrojo integrado de tal manera que esté prendido durante 320  $\mu$ s y medir el valor análogo en el intervalo de 280  $\mu$ s a 320  $\mu$ s, luego el diodo debe permanecer apagado durante 9680  $\mu$ s.

Cuadro I  
SENSORES USADOS PARA EL SISTEMA DE MONITOREO

Medición	Modelo del Sensor	Rango	Unidades
Partículas de Polvo	Sharp GP2Y10	0 a 0.8	mg/m <sup>3</sup>
Dióxido de Carbono	MQ135	10 a 10000	ppm
Temperatura	LM35	-55 a 150	C

Debido a que estos sensores son de bajo costo, se tiene limitaciones al momento de obtener datos muy precisos, sin embargo, en lugares de altas concentraciones estos sensores logran tener un desempeño aceptable, dando una buena referencia de las concentraciones medidas.

Los valores de voltaje emitidos por los sensores de CO2, temperatura y partículas de polvo PM2.5, son transformados internamente dentro del nodo de medición a sus valores equivalentes de partes por millón (ppm), grados centígrados y ug/m3 respectivamente. Posteriormente, estos valores deben ser transmitidos hacia el Gateway, para esto es necesario codificar los datos en una trama serial. La codificación consta de caracteres de control entre los valores medidos por los sensores. Esta trama es luego enviada por el puerto serial para la transmisión inalámbrica por medio del módulo Xbee.

El diagrama de flujo del sketch creado en Arduino se muestra en la Fig. 2 en donde se resume el procedimiento de captura y envío de datos hecho por el nodo de medición. Además en la Fig. 3 se muestra una fotografía de la implementación física del prototipo del nodo de medición.

### II-C. Implementación del Gateway en el Raspberry Pi

Los datos transmitidos por el nodo de medición llegan hacia el gateway implementado en el Raspberry Pi vía Wireless por medio del módulo Xbee para ser interpretados y



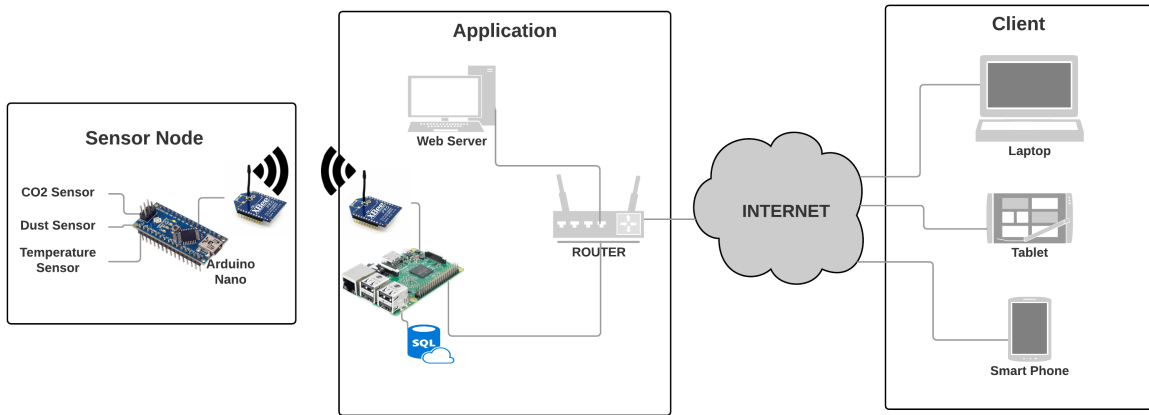


Figura 1. Topología del sistema de monitoreo ambiental implementado

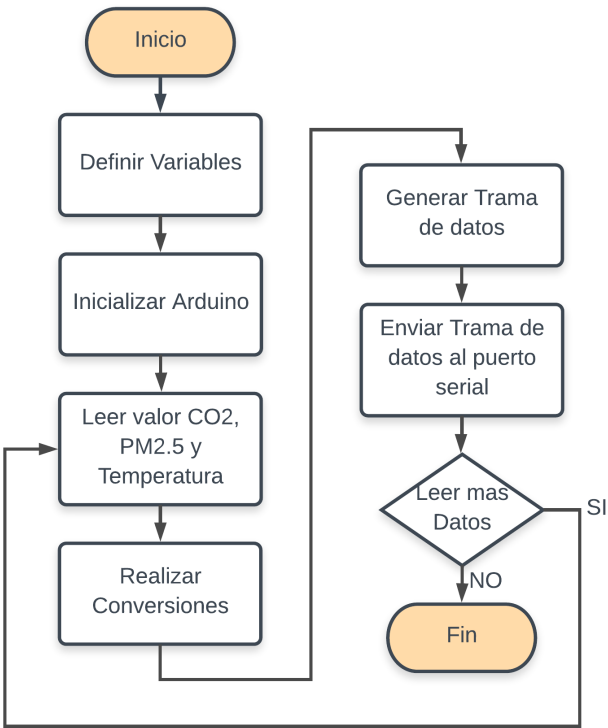


Figura 2. Diagrama de flujo sketch Arduino

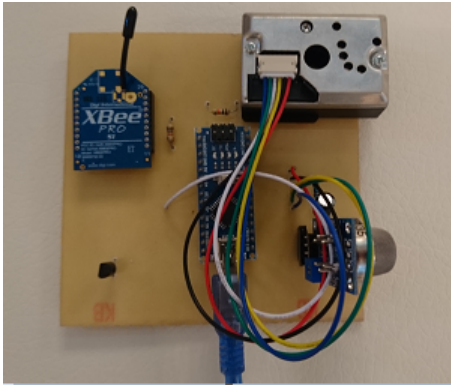


Figura 3. Fotografía de la Placa electrónica con los sensores del prototipo del nodo de medición.

la trama recibida y se separan los caracteres de control que están entre los valores medidos por los sensores.

almacenados en la base de datos. La decodificación de la trama se implementó en el Raspberry Pi mediante un script programado Python 2. Para esto, es necesario que el puerto serial esté configurado para ser utilizado por los puertos digitales exteriores del Raspberry, ya que ahí es donde se conecta el módulo Xbee. La implementación se muestra en la Fig. 4.

El objetivo de la decodificación es separar de la trama de datos recibida los valores medidos por cada sensor para su posterior escritura en la base de datos, en este caso, se toma

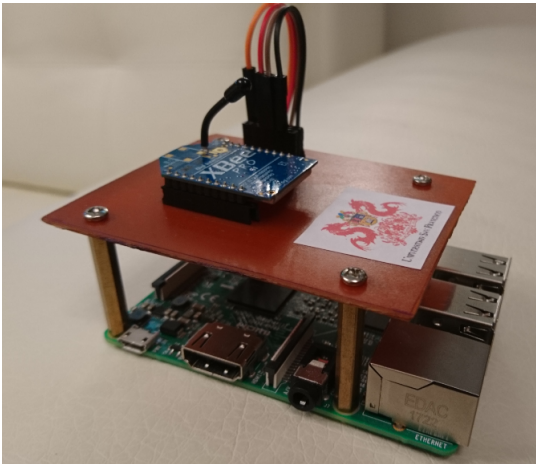


Figura 4. Raspberry Pi con el módulo de comunicación Xbee

#### II-D. Escritura en la base de datos (MariaDB)

En el Raspberry Pi se instaló la base de datos MariaDB 10.1 [13], que es una base de datos tipo MySQL. Se creó la base de datos y dentro de esta una tabla en donde se va a escribir los datos medidos por los sensores. La tabla creada tiene 6 campos que son: día, hora, identificación del nodo de medición, valor de concentración de CO2, valor de concentración de PM2.5, valor Temperatura. En el script implementado se realiza una conexión local con la base de datos, se toman los datos decodificados anteriormente y se los inserta en la base de datos.

El diagrama de flujo del script programado en Python 2 se muestra en la Fig. 5 en donde se resume la decodificación de datos recibidos y escritura en la base de datos.

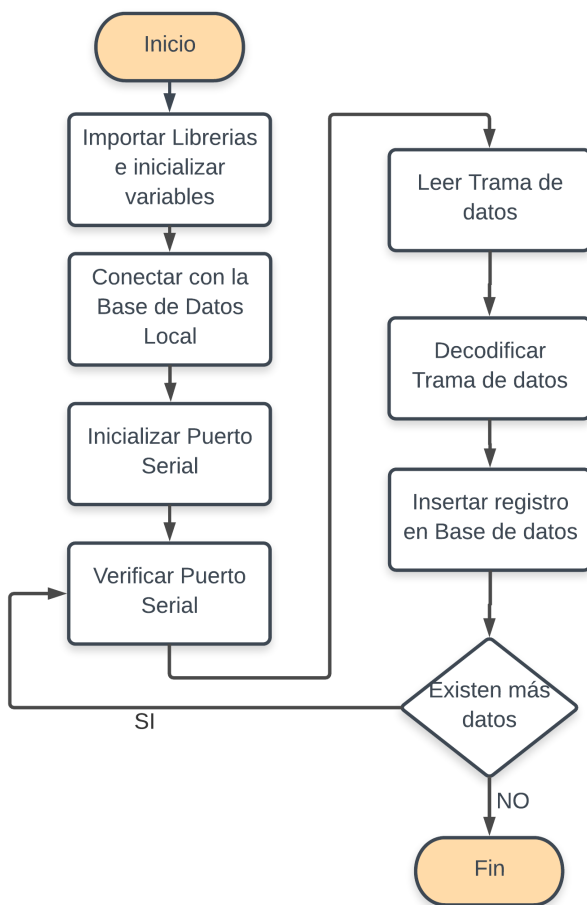


Figura 5. Diagrama de flujo de script en Python implementado en el Raspberry Pi

#### II-E. Implementación del web server

Para poder realizar la visualización en tiempo real por medio de una página web se necesita de un servidor de aplicaciones web, el servidor utilizado en este proyecto fue Apache 2.4.29 [14]. Se realizó la conexión hacia la base de datos ubicada en el Raspberry Pi usando PHP. Para lo cual es necesario que

el Raspberry Pi y el servidor web se encuentren en la misma red local y que la IP del servidor cuente con los permisos necesarios para poder acceder a la base de datos.

Los parámetros necesarios para establecer conexión con la base de datos remotamente son: IP del host, en este caso el Raspberry Pi, el nombre de usuario, la contraseña y el nombre de la base de datos a acceder. Posteriormente, se realiza una instrucción de consulta, que retorna los datos solicitados. En el caso del gráfico en tiempo real, se consulta a la base de datos el valor y la fecha del último dato almacenado. Para el caso del gráfico de mediciones de días anteriores se consulta todos los valores almacenados en la fecha seleccionada por el usuario. Por medio de estas instrucciones en PHP, en el servidor web y definiendo una IP estática en el Raspberry Pi, es posible capturar los datos que se encuentran en la base de datos.

Una vez capturados los datos, estos se deben transformar a un formato que JavaScript pueda interpretar. Este paso se lo logra mediante el comando PHP json encode (JavaScript Object Notation), que convierte un objeto de datos en una cadena de caracteres que JavaScript puede interpretar.

#### II-F. Consultas en tiempo real

Posteriormente, se realizó el diseño de la página web principal en lenguaje HTML y JavaScript. La página web cuenta con cuatro secciones principales que son: cabecera, cuerpo, menú de aplicación y el pie de página. El cuerpo es donde se muestra tres gráficos que corresponden a las mediciones en tiempo real realizadas por los sensores de CO2, Pm2.5 y temperatura. Se utilizó la librería JavaScript de HighChart para graficar los parámetros medidos por los sensores. En este caso, como se trata de un gráfico en tiempo real es necesario utilizar AJAX (Asynchronous JavaScript And XML), que es un método que sirve para aplicaciones en donde se necesite gestionar solicitudes web sin la necesidad de recargar la página. Se solicita la consulta a la base de datos del valor y fecha del último registro almacenado como una coordenada en formato JSON. Las coordenadas devueltas corresponden a la hora y al valor de medición y son graficadas en el eje X y Y respectivamente. En la Fig. 7 se muestra el diagrama de flujo que sigue la página web creada.

Figura 6. Formulario para consultas pasadas

## II-G. Consultas de mediciones pasadas

La aplicación web también cuenta con la opción de consultar mediciones de días anteriores, de la misma manera se utilizó la librería JavaScript de HighChart para poder graficar los datos solicitados. Para esta solicitud se creó una nueva interfaz en PHP, en donde el usuario escoge el parámetro, por ejemplo el tipo de gas, la fecha y el nodo de medición. Dicha interfaz se muestra en la Fig. 6.

Con los datos ingresados por el usuario se realiza la consulta a la base de datos. En este caso, se procede a graficar todos los datos consultados a la vez.

## II-H. Publicación en Internet

Para poder publicar la página web en Internet se creó un dominio web y se configuró el servidor de aplicaciones Apache para que publique en la IP asignada al dominio web. Además, se configuró los puertos de conexión de la página web, para que sea direccionada por el puerto 8000. Se utilizó el programa DUC de NO-IP.com [14] para gestionar el dominio y actualizar la IP remota del servidor web.

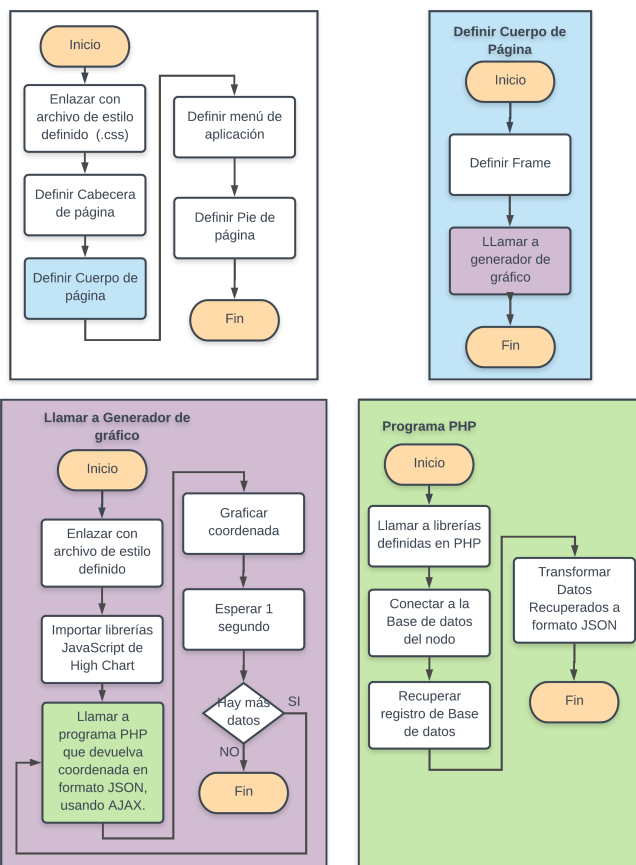


Figura 7. Diagrama de Flujo página web

## III. RESULTADOS

La interfaz gráfica que se presenta al usuario se muestra en la Fig. 8, en donde se puede observar el resultado final

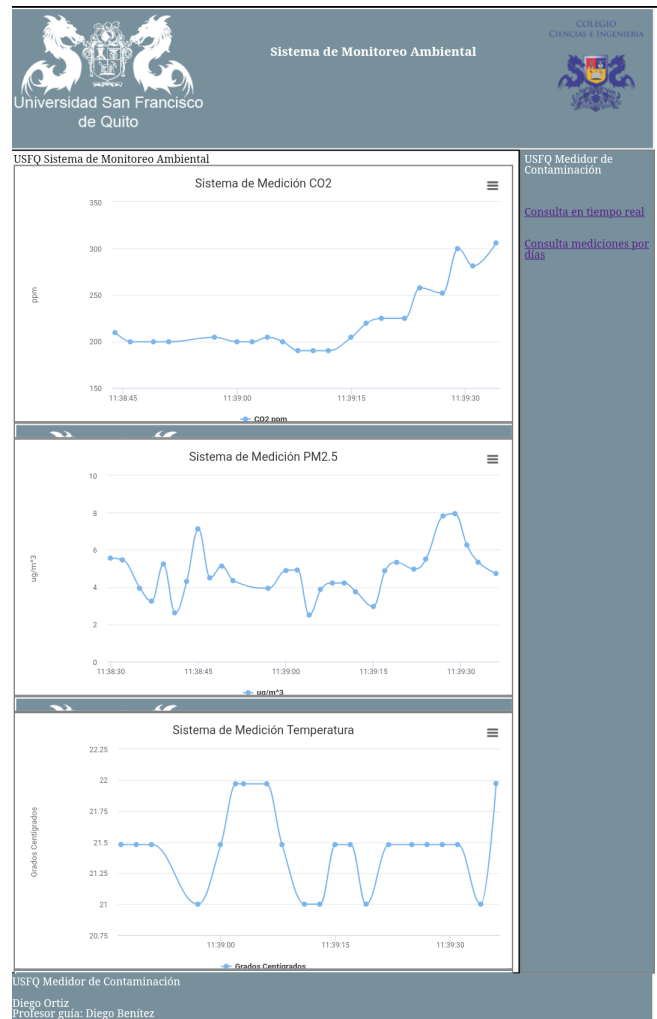


Figura 8. Captura de pantalla de la página web implementada

de la página web mostrando los resultados en tiempo real. Es importante recalcar que, con la metodología presentada, el servicio web se encuentra disponible en Internet y para cualquier plataforma de visualización. En el lado del cliente el programa de JavaScript es ejecutado, mientras que la conexión y autenticación hacia la base de datos se la hace localmente con las instrucciones PHP en el servidor.

Para comprobar el correcto funcionamiento del prototipo implementado, se realizaron varias mediciones en las instalaciones de la Universidad San Francisco de Quito, en Cumbaya. Los lugares donde se realizaron mediciones fueron en los exteriores del campus y dentro de un parqueadero ubicado en subsuelo 5. Las mediciones fueron tomadas cada 2 segundos durante 5 horas consecutivas. Los resultados obtenidos se muestran en la Fig. 9 en donde se muestra la concentración de CO2 en ambos lugares. Como es de esperar, la concentración dentro del parqueadero subterráneo supera a la concentración en aire libre. La concentración promedio dentro del parqueadero subterráneo es de 1097ppm con una desviación estándar de 207.70, mientras que la concentración promedio al aire

libre es de 316.77 con una desviación estándar de 50.15. De acuerdo con Kane International [7] la concentración presente de CO<sub>2</sub> al aire libre está entre 250 y 350ppm mientras que a nivel atmosférico se encuentra alrededor de 400ppm [8]. Los valores obtenidos para exteriores se encuentra dentro de este rango referencial, mientras que la concentración en el parqueadero subterráneo esta dentro del rango de mal aire [7].

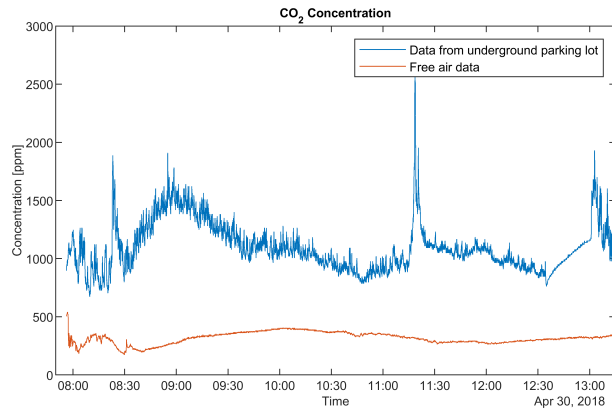


Figura 9. Mediciones de CO<sub>2</sub> dentro del parqueadero subterráneo y al aire libre

En la Fig.10 se muestran los datos obtenidos de PM<sub>2.5</sub>. Como se puede observar nuevamente, las concentraciones de partículas de polvo son mayores en los interiores del parqueadero. El valor promedio de concentración dentro del parqueadero subterráneo es de 13.59  $\mu\text{g}/\text{m}^3$  con una desviación estándar de 3.80, mientras que la concentración en aire libre es de 6.85  $\mu\text{g}/\text{m}^3$  con una desviación estándar de 3.04. Según la secretaria de ambiente de Quito [8], en el mes de abril de 2018 el promedio mensual en Tumbaco fue de 12.09  $\mu\text{g}/\text{m}^3$ . Finalmente se obtuvieron los valores de temperatura en ambos lugares. El valor promedio para el parqueadero fue de 26.42°C mientras que para los exteriores de 22.40°C.

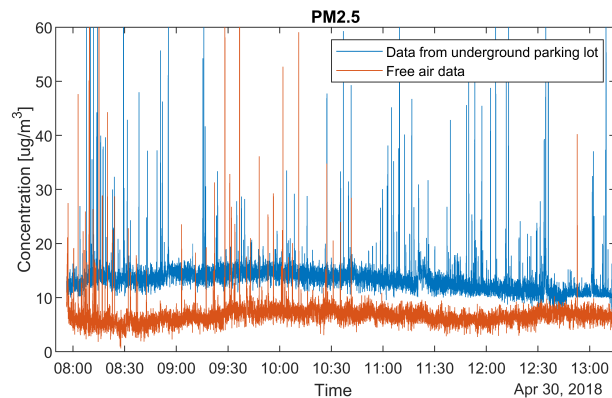


Figura 10. Mediciones de PM<sub>2.5</sub> dentro del parqueadero subterráneo y al aire libre

#### IV. CONCLUSIÓN

En este proyecto se diseñó e implementó un sistema de monitoreo ambiental de bajo costo basado en redes de sensores

inalámbricos (WSN). Con esta topología se evita que los datos se pierdan cuando no haya una conexión a Internet o de red, además es una topología expandible ya que se puede aumentar el número de sensores para diferentes gases, para el aumentar el área de monitoreo.

Los nodos de medición cuentan con un Arduino Nano, sensor de CO<sub>2</sub>, PM<sub>2.5</sub>, temperatura y el módulo de comunicación Wireless Xbee. Los datos medidos por los sensores son codificados y enviados vía Wireless hacia el Raspberry Pi, en donde se decodifica la trama y se almacena la base de datos. Una aplicación web fue creada para poder visualizar los datos medidos por los sensores en tiempo real y mediciones pasadas. Esta aplicación web es publicada en Internet disponible para cualquier dispositivo. Pruebas iniciales demostraron que el sistema funciona adecuadamente, se obtuvieron mediciones ambientales en Cumbayá dentro y fuera de un parqueadero subterráneo en donde las concentraciones de los gases se encontraban dentro del rango de medición de los sensores de bajo costo. Para trabajos futuros se puede aumentar el número de nodos, usar sensores más precisos y experimentar con odroid como gateway.

#### REFERENCIAS

- [1] World Health Organization: 7 million deaths in 2012 due to air pollution. 2014.
- [2] Kumar, D. Katoria. "Air Pollution and its Control Measures". International Journal of Environmental Engineering and Management. ISSN 2231-1319, Volume 4, Number 5 (2013), pp 445-450.
- [3] Choi, S., Kim, N., Cha, H., Ha, R., (2009). Micro Sensor Node for Air Pollutant Monitoring: Hardware and Software Issues. Sensors. ISSN 1424-82, pages 7970 - 7987
- [4] Raja Vara Prasad, Mirza sami Baig, Rahul K. Mishra, P. Rajalakshmi, U.B. Desai, S.N. Merchant, (2011). "Real Time Wireless Air Pollution Monitoring System". ICTACT Journal on Communication Technology: Special Issue on Next Generation Wireless Networks and Applications, volume - 2, Issue - 2. ISSN: 2229-6948
- [5] Devarakonda, Srinivas, et al. Real-time air quality monitoring through mobile sensing in metropolitan areas." Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing. ACM, 2013.
- [6] Simaitis, R., Baltrenas, P. "Concentration of air contaminants in Vilnius and its impact on human health". Environmental Engineering, Vol X, No 1 Vilnius: Technika
- [7] Kane International. "What are safe levels of CO and CO<sub>2</sub> in rooms?". [Online] Available: <https://www.kane.co.uk/knowledge-centre/what-are-safe-levels-of-co-and-co2-in-rooms>
- [8] [7] NASA, Global Climate Change. 2018. Available online <https://climate.nasa.gov/>
- [9] Secretaria de Ambiente Quito (2018). Reporte mensual de Abril de 2018 de CO<sub>2</sub>.
- [10] NO-IP.com. Dynamic DNS Update Client (DUC) for Windows. [Online]. Available: <https://www.noip.com/download>
- [11] Digi International. Xbee/Xbee-PRO S1 802.15.4 (Legacy). [Online]: Available: <https://www.digi.com/resources/documentation/digidocs/pdfs/90000982.pdf>
- [12] Atmel. 8-bit AVR Microcontrollers Atmega328P. [Online]: Available: [http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P\\_Datasheet.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf)
- [13] MariaDB. What is it? [Online] Available: <https://mariadb.com/sites/default/files/MariaDB.pdf>
- [14] Apache Software Foundation. Apache HTTP Server Documentation Version 2.4. 2015. [Online] Available: <https://archive.apache.org/dist/httpd/docs/httpd-docs-2.4.16.en.pdf>