2019

Análisis y Monitoreo de la calidad del aire en áreas Metropolitanas

Terradas, Matias Santiago

UCASAL – Facultad de Ingeniería

01/01/2019

Titulo

Ingeniería en Informática

Profesor Guía

Nombre: Ing. Pablo Sebastian Vargas **Firma:** _____

Tribunal Evaluador

Nombre: _____ **Firma:** _____

Nombre: _____ **Firma:** _____

Nombre: _____ **Firma:** _____

Fecha de Exposición

_____/_____/_____

Agradecimientos

En primer lugar quiero agradecer a mis padres, Lillian y Luis, por su inmenso amor, esfuerzo, apoyo y comprensión durante todo este tiempo, sin ellos no hubiese sido posible.

A mi hermana Melina, por ser una excelente hermana mayor y estar siempre pendiente de mí.

A mi familia, mi novia Pura y amigos de toda la vida. Por acompañarme, apoyarme siempre y por todo el cariño.

A mi director de tesis Ing. Pablo Sebastián, Vargas, por estar siempre predispuesto a ayudarme.

Al Lic. José Peralta, por la ayuda brindada a lo largo de la carrera y por su vocación de docente, siempre predispuesto a enseñar, comprender y ayudar a sus alumnos.

A todo el personal de la universidad con el cual compartí durante estos años. Entre ellos docentes que inspiran e incentivan a seguir aprendiendo siempre. A Patricia que siempre esta predispuesta a ayudar a todo aquel que se acerca a secretaria.

A todos mis compañeros y amigos con los cuales compartí gratos momentos tanto dentro como fuera de la universidad, en especial Maximiliano y Mariano.

Contenido

Agradecimientos 2
Contenido 3
Abstract 9
1-Introduccion 10
Breve descripción del problema 10
Carácter del problema 10
Motivación para abordarlo resaltando la importancia de resolverlo 10
Pasos que se realizaran 11
Criterios de éxitos 11
2-Estado de la cuestión 12
Fundamentos teóricos y prácticos que sustentan las soluciones escogidas 12
Elementos básicos de IoT 13
Hardware 13
Infraestructura 14
Software 14
Tecnologías para desarrollo de IoT 14
Plataforma 15
Sensores 16
Medio de comunicación 16
Antecedentes relacionados con el problema 18
Técnicas y herramientas que fueron requeridos en el tratamiento del problema 19
Alternativas de Hardware 24
Temas pendientes de resolución 25
3-Definición del problema 26
Objetivos 26
Requisitos 26
Requisitos de Hardware 29

Requisitos de Software 30
Restricciones 30
Alcance 30
Análisis FODA 31
4-Solución propuesta 35
Análisis de Factibilidad 35
Factibilidad Económica 35
Beneficio Económico Esperado 39
Factibilidad Técnica 39
Factibilidad legal 40
Gestión de Riesgos 40
Análisis comparativo de conectividad 41
¿Qué es LoRa y porque se usó? 42
¿Qué es LoraWAN? 44
Procedimiento lógico del Sistema 45
Hardware 47
Software 56
Herramientas Utilizadas 70
Apache 70
MySQL 70
XAMPP 70
Laravel Lumen 70
PHP 70
PHPStorm 71
JQuery 71
Bootstrap 71
Postman 71
Putty 71

Git 72
Waziup 72
Lora Framework 72
Fritzing 72
Metodología 72
5- Resultados o verificación experimental 81
Experimento y prueba realizada 81
Hardware 81
Software 83
Cálculo de consumo del circuito 85
Resultados obtenidos 87
Dificultades no previstas encontradas y sus soluciones 94
6- Conclusiones y futuras líneas de investigación 96
7- Bibliografía 97
7- Anexos 98
Anexo 1 Instalación y configuración del nodo Gateway 98
Anexo 2 Configuración Software de nodo Sensor 104
Anexo 3 Instalaciones y configuraciones del servidor 107

Índice de Figuras

Figura 1 - IoT 12
Figura 2 - Arduino MINI PRO 15
Figura 3 - Sensores 16
Figura 4 - Medios de Comunicación para IoT 17
Figura 5 - Estación EPA 18
Figura 6 - Arduino Mini Pro 20
Figura 7 - Raspberry Pi 3 Zero 20
Figura 8 - Modulo Transceptor Sx1278 21
Figura 9 - Modulo Sensor BMP280 21
Figura 10 - Sensor DHT11 22
Figura 11 - Modulo Sensor de gas MQ-9 23
Figura 12 - Fuente de alimentación Protoboard 23
Figura 13 – FODA 32
Figura 14 – LoRa 42
Figura 15 - cuadro de atribución de bandas de frecuencias de la República Argentina – Enacom 44
Figura 16 - Diagrama simplificado 45
Figura 17 - Ejemplo ilustrativo de topología de red 47
Figura 18 - Nodo Gateway 48
Figura 19 - Diagrama de circuito nodo Gateway 50
Figura 20 - circuito con componentes ilustrados - nodo Gateway 51
Figura 21 - Nodo Sensor 53
Figura 22 - PCB Circuito nodo Sensor 54
Figura 23 - Esquema con componentes ilustrados - nodo Sensor 55
Figura 24 - Diagrama de flujo del sistema 57
Figura 25 - Pantalla de Login 58
Figura 26 - Conexión remota mediante SSH a un nodo Gateway 59
Figura 27 - Pantalla de detalle nodo Sensor Rotonda 60
Figura 28 - Pantalla de detalle del nodo sensor 61
Figura 29 - Últimos datos registrados por el sensor 62
Figura 30 - Pantalla dashboard (menú principal) 63
Figura 31 - Formulario emergente para editar datos del nodo 64
Figura 32 - Lista de nodos Gateways 65
Figura 33 - Configuración de parámetros arduino 67
Figura 34 - Salida por puerto serie de la ejecución del software en arduino 68
Figura 35 - Datos enviados al servidor desde el nodo Gateway 69
Figura 36 - Metodología Iterativa e Incremental 73

Figura 37 - Diagrama de actividades 75
Figura 38 – Diagrama caso de uso 1 76
Figura 39 - Diagrama de clases conceptuales 77
Figura 40 - DER 79
Figura 41 - Diagrama caso de uso 2 80
Figura 42 - Resultado de prueba de alcance modulo LoRa 82
Figura 43 - Prueba de dirección de antena en modulo LoRa 83
Figura 44 - Pruebas de API REST con POSTMAN. 84
Figura 45 - Herramienta de depuración de Google Chrome 85
Figura 46 - Oscilación de CO en una semana 88
Figura 47 - Relación CO y temperatura 89
Figura 48 - Zoom en el gráfico de Temperatura y CO 90
Figura 49 - Niveles de CO y Temperatura - Brasil 91
Figura 50 - Temperatura y presión atmosférica salta 92
Figura 51 - Calculadora ICA 93
Figura 52 - Cables monofilamento para protoboard 95
Figura 53 - Instalación ISO 99
Figura 54 - Interfaz web configuración Raspberry Pi Gateway 100
Figura 55 - Conexión SSH mediante PUTTY 101
Figura 56 - Menú de configuración Gateway 103
Figura 57 - Código fuente arduino - Nodo Sensor 104
Figura 58 - Indicar frecuencia del nodo sensor 105
Figura 59 - Configuración de dirección, modo y tiempo 106
Figura 60 - Cambiar clave de aplicación 107
Figura 61 - Clonando repositorio tesis_api 108
Figura 62 - Crear nueva base de datos mysql 109
Figura 63 - Importando la base de datos 109
Figura 64 - Configuración de archivo .env 110
Figura 65 - panel de control xampp 111
Figura 66 - Pantalla de acceso 112
Figura 67 - Página principal 113
Figura 68 - operaciones de marcador 114
Figura 69 - formulario de edición 115
Figura 70 - Detalle de nodo sensor 116
Figura 71 - control de desplazamiento 117
Figura 72 - Menú de opciones - >lista 118
Figura 73 - Listado de gateways 118
Figura 74 - Menú de opciones 119

Índice de Tablas

Tabla 1 - Hardware alternativo 24
Tabla 2 - Rangos en los que han sido clasificados los valores del ICA de acuerdo 27
Tabla 3 - Acciones preventivas dependiendo del valor del ICA para el 28
Tabla 4 - Punto de corte para el CO para el ICA 29
Tabla 5 - Costos de Hardware 37
Tabla 6 - Costos de RRHH 37
Tabla 7 - Costos de Equipamiento 38
Tabla 8 - Costos de Servidor 38
Tabla 9 - Costos de DNS 38
Tabla 10 - Total de costos del proyecto 38
Tabla 11 - Costos por Nodos 39
Tabla 12 - Tabla de Riesgos 41
Tabla 13 - Comparativa de Tecnologías 42
Tabla 14 - Conexiones modulo LoRa a Raspberry pi 52
Tabla 15 - Conexión Arduino Modulo LoRa 56
Tabla 16 - Parámetros del nodo Sensor 66
Tabla 17 - LoRa Mode: Variación de combinación de ancho de banda y dispersión de espectro 67
Tabla 18 - Estructura del paquete 68
Tabla 19 - Consumo de componentes del circuito 86
Tabla 20 - Tipos de fuentes y capacidades 86
Tabla 21 - Posibles errores en el nodo Sensor 94

Abstract

Tal como informa la Organización Mundial de la Salud, la contaminación del aire representa un importante riesgo medioambiental para la salud. Mediante la disminución de los niveles de contaminación del aire los países pueden reducir la carga de morbilidad derivada de accidentes cerebrovasculares, cánceres de pulmón y neumopatías crónicas y agudas, entre ellas el asma.

Este proyecto de grado busca crear una red de sensores que tiene como fin brindar una herramienta para generar conciencia del cuidado del medio ambiente, previendo una solución de bajo presupuesto comparado con otras soluciones existentes. Si bien no es algo que soluciona el problema de raíz, brinda información necesaria para que sea usada y promovida con fines benévolos que ayuden a tomar decisiones para mejorar la calidad de vida de las personas y a preservar el medio ambiente.

1-Introduccion

Breve descripción del problema

La principal problemática es que en el País no contamos con redes de este tipo. En nuestra ciudad particularmente se presenta la problemática del crecimiento del parque automotor en este último tiempo, con lo cual las emisiones de gases contaminantes han aumentado y no se tiene ningún control sobre estas. Otro problema adicional es el costo que tienen estas estaciones. Al no contar con estas tecnologías no estamos siendo conscientes de la realidad ambiental en la que vivimos ni podemos realizar simulaciones para saber cómo podría ser la situación en unos años.

Carácter del problema

El problema es de carácter social, ya que la calidad de vida es de interés público y general. El cuidado y la preservación del medio ambiente es algo que nos compete a todos como sociedad.

La creciente preocupación por el cambio climático demanda soluciones de este tipo para vigilar el medio ambiente, permitiendo así, un desarrollo sostenible de las ciudades y una mejora de la calidad de vida humana.

Motivación para abordarlo resaltando la importancia de resolverlo

El principal motivo es la facilidad y ventaja que proveen las diferentes soluciones de IoT que existen en el mercado actual (Arduino, Raspberry Pi, gran variedad de módulos y sensores como así también protocolos). Hoy en día existen diversas herramientas y tecnologías que permiten que proyectos que antes demandaban gran cantidad de recursos y conocimiento, puedan ser implementados con una reducción considerable de personal, tiempo y dinero.

Por otro lado surge la motivación de buscar una solución más económica para que pueda ser implementada con mayor facilidad, instalando mini-estaciones de control en diversos puntos de la ciudad, teniendo así no una única central costosa en un determinado punto, sino varias mini-estaciones de menor presupuesto que permitan conseguir resultados de calidad.

La importancia de resolver este problema no es menor, día a día el parque automotor sigue creciendo y con ello las emisiones de CO2, entre otros agentes contaminantes producto las industrias, mal manejo de residuos y consumo excesivo.

Pasos que se realizaran

Lo primero a realizar es una investigación referente a la calidad esperada del aire en una ciudad no contaminada para que a partir de esos parámetros se pueda referenciar y saber así como determinar la calidad del aire. Dicha investigación estará sujeta a todo lo referido por la OMS (Organización Mundial de la Salud).

Lo siguiente es analizar qué alternativas de hardware existen en el mercado, realizando una comparación de costos y beneficios. También se analizaran los distintos medio de comunicación para montar la red y los protocolos que existen para realizar las comunicaciones entre las estaciones (a partir de ahora nodos sensor) y la central (nodo Gateway).

Se analizara que arquitectura de red conviene más para aplicar a la solución.

Cuando ya estén definidos todos los requerimientos anteriormente mencionados, se procederá a la construcción de un prototipo a fines de demostrar una prueba de concepto de la solución propuesta y su funcionamiento. El prototipo contara con el hardware necesario para su funcionamiento y el software que se encargara de manipular y presentar los datos.

Criterios de éxitos

Se definen como criterios de éxito:

- Construcción de prototipo funcional
- Recolección de datos de calidad
- Prueba del prototipo para controlar que es funcional
- Presentación de los resultados obtenidos mediante una plataforma web

2-Estado de la cuestión

Fundamentos teóricos y prácticos que sustentan las soluciones escogidas

INTERNET DE LAS COSAS: Internet en sus inicios fue desarrollado con el objetivo de compartir únicamente información, al pasar de los años progresivamente ha evolucionado permitiendo mejorar los servicios que ofrece al usuario desde prestar servicios de streaming de voz y de video hasta realizar transacciones bancarias desde la comodidad del hogar.

Figura 1 - IoT

Fuente: http://ebimcorp.biz – 2017

El término IoT, traducido como Internet of Things, define cualquier objeto o cosa con capacidad de conectarse a la red e interactuar con otro dispositivo o persona, teniendo para el año 2014 alrededor de dos mil millones de dispositivos conectados, excluyendo teléfonos inteligentes *(Bliznakoff del Valle, 2014 pág. 7)*.

El impacto económico que tienen los proyectos de IoT han impulsado que grandes empresas se involucren en esta nueva tendencia, tomando en cuenta que las primeras empresas involucradas con este tipo de proyectos prevén ingresos de un trillón de dólares en el 2020 gracias a IoT, presentando alrededor de 50 billones de equipos conectados a la red *(Bliznakoff del Valle, 2014 pág. 6).*

La idea fundamental de IoT es dotar de electrónica a cualquier objeto cotidiano y permitir su conexión a Internet, para cumplir este objetivo es necesario una tecnología capaz de mantener la conexión sin importar el lugar. Es aquí donde las tecnologías inalámbricas son la clave fundamental para que cosas u objetos inteligentes puedan comunicarse entre ellos y hacia el Internet, convirtiendo a las Redes de Sensores Inalámbricos en la base del desarrollo de IoT.

Su acogida es debido principalmente a:

- Popularización de placas de hardware libre
- Reducción de costo de sensores
- Plataformas de IoT

Elementos básicos de IoT

Desde un punto de vista estructural, la evolución de IoT no habría podido llevarse a cabo sin la concurrencia de tres áreas concretas que han permitido el desarrollo continuo de cada una de las aplicaciones desarrolladas para plataformas IoT.

Hardware

Los dispositivos fabricados en los inicios de la electrónica ocupaban espacios excesivamente grandes para la función que desempeñaban, en la actualidad la tendencia de la miniaturización de componentes electrónicos ha permitido fabricar dispositivos de tamaño reducido con capacidades similares e incluso superiores a la de una computadora personal.

Infraestructura

La demanda creciente de puntos de acceso hacia Internet y el número de dispositivos que requieren dicha conexión, obligan el desarrollo de nuevas tecnologías que mejoren la gestión de recursos, en este caso el espacio radioeléctrico, para eliminar la saturación de infraestructuras inalámbricas sin degradar el servicio que presta, o buscar alternativas de conexión que mejoren el rendimiento como son la red WI-FI, la red de telefonía móvil e incluso la conexión de fibra óptica que al trabajar en conjunto pueden solventar las limitaciones que presenten cada una de ellas individualmente.

Software

El hardware por sí solo no es capaz de hacer uso de la información generada a partir del IoT, es necesario contar con algoritmos y aplicaciones correctamente diseñadas que permitan interpretar la información recopilada de los dispositivos del IoT y tomar decisiones cuando sea necesario sin la interacción con el ser humano, he ahí la inteligencia de IoT y quizá el componente principal.

Tecnologías para desarrollo de IoT

El objetivo principal de IoT es mantener conectados la mayor cantidad de objetos hacia la red y entre ellos, para lo cual es necesario contar con tecnologías que nos permitan llevar a cabo esta interconectividad, partiendo con las características que deben cumplir los diferentes dispositivos, como puede ser tamaño, peso, periféricos; e incluso el medio por el cual se van a comunicar entre ellos.

Plataforma

Actualmente en el mercado la mayor parte de las industrias desarrollan plataformas enfocadas al IoT, gracias a los avances tecnológicos es posible diseñar equipos de reducido tamaño con capacidades de procesamiento limitadas pero suficientes para cumplir con el objetivo de IoT. Existen empresas dedicadas únicamente a generar dispositivos enfocados al IoT entre los que se encuentran MEMSIC, fabricante de MICAz y TelosB.

A pesar de estas empresas estar enfocadas exclusivamente en plataformas IoT, el alcance al público se puede ver un poco limitado, ya sea por el factor económico o porque en el mercado existen plataformas abiertas con gran soporte y respaldo por la comunidad como es el caso de RASPBERRY o ARDUINO, este último con su placa de desarrollo de mayor aceptación mostrada en la Figura 2 que, si bien en un principio ambas plataformas no se encontraban enfocadas al IoT, han sabido adaptarse para el desarrollo de este tipo de proyectos.

Figura 2 - Arduino MINI PRO

Fuente: https://www.flickr.com/photos/mellis/4784333369 – 2019

Sensores

La plataforma por si sola se encarga únicamente del procesamiento de información, pero esta información es proveniente de dispositivos externos a la plataforma, denominados sensores. Estos dispositivos sensores están diseñados para capturar información de parámetros específicos del entorno donde se encuentren.

La electrónica actual ha permitido abaratar los costos de fabricación de cada uno de estos sensores, permitiendo encontrar un amplio catálogo de estos accesorios para cualquiera que sea nuestro requerimiento a un precio accesible, como los mostrados en la Figura 3.

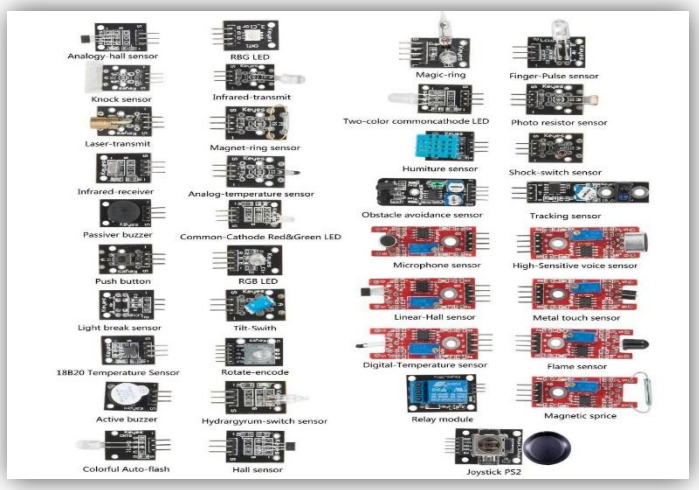


Figura 3 - Sensores

Fuente: https://patagoniatec.com.ar – 2017

Medio de comunicación

Las tecnologías mencionadas en el apartado anterior permiten obtener información del entorno y procesarla de manera limitada, pero si existiese la necesidad de un procesamiento exhaustivo o mucho más rápido es necesario transportar la información recolectada a un equipo con características superiores. Aquí juega un papel importante el canal de comunicación ya que debe ser acorde a las características

de la plataforma en uso, existiendo en la actualidad diferentes tecnologías, mostradas en la Figura 4, que pueden ser utilizadas con este objetivo.

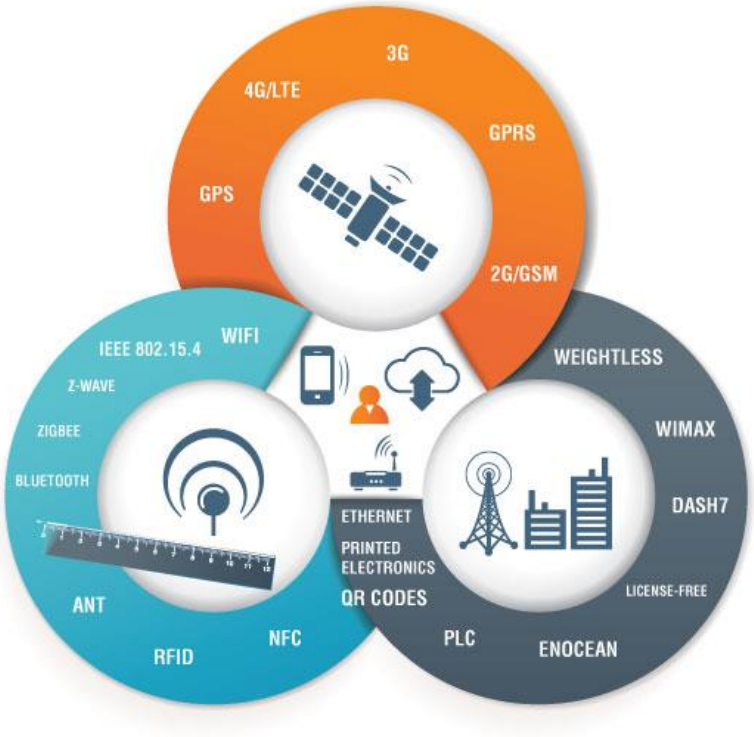


Figura 4 - Medios de Comunicación para IoT

Fuente: :(Potscapes, 2017)

Puede utilizarse medios de comunicación tradicionales como el cable vía Ethernet, pero al tener como base del IoT tecnologías inalámbricas la tendencia es utilizar protocolos inalámbricos que al mismo tiempo posean un bajo consumo energético; para lo cual se han desarrollado tecnologías como Zigbee, 6LowPan, LoRa o Bluetooth 4.0 que permiten extender el tiempo de operación de cada una de las plataformas.

Antecedentes relacionados con el problema

A nivel nacional se encuentran trabajos de este tipo en la ciudad de Buenos Aires donde se instaló una estación para monitoreo de la calidad del aire. Ubicada en área residencial-comercial con flujo vehicular medio y muy escasa incidencia de fuentes fijas. Es una estación EPA totalmente automática emplazada en cabina transportable.

Figura 5 - Estación EPA

Fuente: http://www.buenosaires.gob.ar/ -2017

¿Qué mide esta estación de monitoreo atmosférico?

Contaminantes Atmosféricos:

- CO
- NO, NO2 y NOx
- Material Particulado (PM10)

Parámetros Meteorológicos Asociados:

- Temperatura
- Humedad
- Presión atmosférica
- Velocidad y dirección del viento
- Radiación solar
- Precipitaciones

A nivel mundial podemos encontrar WAQI, un proyecto que busca informar en tiempo real la contaminación del aire a nivel mundial. Cuenta con más de 10.000 estaciones alrededor del mundo, el proyecto comercializa las estaciones para todo aquel que quiera participar.

Técnicas y herramientas que fueron requeridos en el tratamiento del problema

Se utilizaron diversas tecnologías tanto de software como de hardware para desarrollar la solución.

Para almacenar los datos recolectados se empleó el motor de base de datos **MySQL** por ser libre. En cuanto a los lenguajes de programación se usaron **PHP** para el aplicativo web donde se presentaran los datos recolectados, el cual se escoge por la amplia trayectoria y aplicación en desarrollo web, en particular se agilizo el desarrollo mediante el uso del **framework Laravel** que es actualmente el más usado en el mercado dado a su robustez y los beneficios de contar con una amplia y activa comunidad. También se emplearon otros lenguajes a fin de enriquecer el aplicativo web, como ser **JavaScript, CSS**.

En cuanto al hardware necesario tenemos requerimientos de placas controladoras, módulos y sensores a los cuales se les hizo mención anteriormente pero que se listan con más detalle a continuación:

Arduino Mini Pro: Es la versión de arduino más pequeña, la idea es que cada nodo sensor tenga un tamaño reducido para así también disminuir el consumo energético y el coste de construcción. Dentro de cada nodo sensor encontramos uno de estos el cual se encargara de controlar los sensores y enviar los datos recolectados al nodo Gateway.

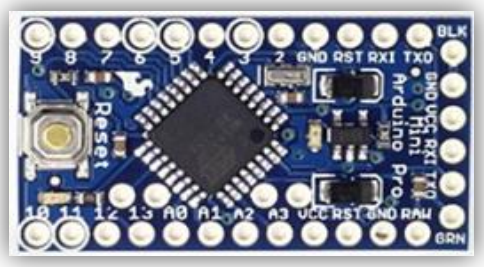


Figura 6 - Arduino Mini Pro

Fuente: https://store.arduino.cc/ -2019

Raspberry Pi: Es una mini computadora literalmente, ya que cuenta con microprocesador, placa madre y memora RAM. Es necesaria para crear lo nodo Gateway, que como se mencionó anteriormente se encarga de recibir los datos de los nodos Sensores y enviarlos al servidor en la nube para que sean visualizados en el aplicativo web.

Figura 7 - Raspberry Pi 3 Zero

Fuente: https://raspberry.org/ -2019

Módulo Transceptor Lora1278 Sx1278 Rf: Este es un pequeño módulo de radio frecuencia capaz de enviar y recibir datos, se caracteriza por ser de bajo consumo y largo alcance (tecnología que se conoce como LoRa). Es el encargado de realizar la comunicación entre los diferentes nodos. En lugares descampados o con poca interferencia alcanza sin problemas los 4km de distancia.

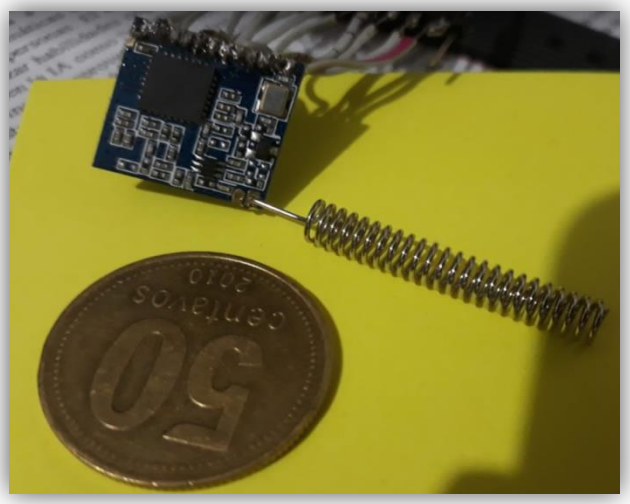


Figura 8 - Modulo Transceptor Sx1278

Módulo Sensor de Presión y Temperatura BMP280: El sensor BMP280 integra en un solo dispositivo sensores de presión atmosférica, temperatura y humedad relativa, con gran precisión, bajo consumo energético y un formato ultra compacto. Permitirá recolectar datos como la temperatura y presión atmosférica que podría servir para realizar minería de datos en el futuro, por ejemplo ver si existe alguna relación entre los niveles de contaminación y temperatura del aire o analizar si incide o no de alguna manera.

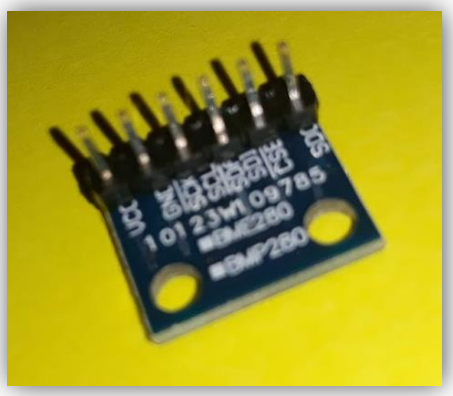


Figura 9 - Modulo Sensor BMP280

Módulo Sensor de Presión y Temperatura DHT11

Sensor con una alta fiabilidad y estabilidad debido a su señal digital calibrada. Funciona con un voltaje de entre 3,5 y 5 voltios. Permite medir rangos de temperatura de entre 0° y 50° y de humedad 20% a 90%.

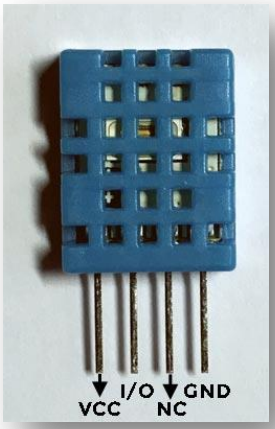


Figura 10 - Sensor DHT11

Fuente: https://programarfacil.com 2019

Módulo Sensor de gas MQ-9: El MQ-9 puede detectar concentraciones de CO de 10 a 10000 ppm (partes por millón) y de 100 a 10000 ppm de gas combustible. Es empleado para tomar mediciones del monóxido de carbono presente en el aire.

Figura 11 - Modulo Sensor de gas MQ-9

Fuente de alimentación para protoboard: Es una fuente que permite convertir la corriente alterna en corriente continua para poder suministrar de energía a una protoboard donde se diseña y experimentan los circuitos electrónicos antes de ser ensamblados y soldados. Se puede conectar mediante USB y cuenta con dos canales de alimentación intercambiables entre 3.3V y 5V. También se puede conectar al 220V por medio de una fuente de 9V o 12v.

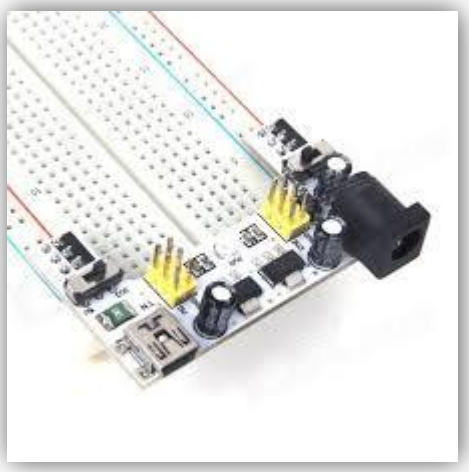


Figura 12 - Fuente de alimentación Protoboard

Alternativas de Hardware

En caso de no poder adquirir algunos de los componentes de hardware mencionado anteriormente, se plantea la siguiente tabla de posibilidades:

Hardware Requerido	Hardware Alternativo	Implica Mayor Costo	Provee mejor funcionamiento
Arduino Mini Pro	- Arduino Nano - Arduino Uno - Arduino Mega	SI	Relativamente si ya que estos modelos poseen más poder de procesamiento aunque no se requiere mucho.
Raspberry Pi Zero	- Raspberry Pi 2 o 3 Model b	SI	Si ya que se podrían controlar un mayor número de nodos sensores.
Módulo Transceptor Lora1278 Sx1278 Rf	- Módulo Transceptor Lora Um402	SI	No, pero si probé mayor facilidad a la hora de armar el circuito.
Módulo Sensor de Presión y Temperatura BMP280	- Módulo BME 280	NO	Sise A diferencia del BMP280, el BME280 mide humedad, es más preciso y no se requerirá el DHT11.
Módulo Sensor de Presión y Temperatura DHT11	- DHT22 (AM2302)	SI	Relativamente si, es un poco más preciso y permite un rango mayor de temperaturas.
Módulo Sensor de gas MQ-9	- Módulo sensor MQ-7	NO	No, el sensor que se escogió es más sensible que el alternativo.

Tabla 1 - Hardware alternativo

Desarrollo de una aplicación web: Se desarrolló un aplicativo web que sirve para almacenar los datos recolectados por los nodos sensores, en una base de datos y ser finalmente presentados ante los usuarios.

Temas pendientes de resolución

El presente proyecto no abarca la totalidad del desarrollo de la solución propuesta por lo que se dejara de lado, algunos aspectos que están fuera del alcance académico pretendido. Entre estos tenemos:

1. **Topología de red mesh (tipo malla):** Permite la interconexión de múltiples nodos entre si, pudiendo comunicar datos entre si hasta llegar al nodo Gateway, lo que permitirá extender el alcance de la red sin necesidad de adquirir más nodos Gateway y así también abaratar costos.
2. **Alimentación con energía solar:** Ya que se trata de un proyecto ecológico, lo ideal es que los nodos puedan abastecerse de energías renovables como la energía solar, mediante un panel solar que suministre corriente a los circuitos.
3. **Sensor de partículas:** Es deseable analizar la mayor cantidad de factores que determinan la calidad del aire, entre ellos, la cantidad de partículas presentes es de importancia para el análisis.
4. **Incorporar nuevos sensores:** Es de suma importancia la incorporación de más señores que permitan analizar otros factores que contaminan el aire cuando su presencia supera un determinado valor.

3-Definición del problema

La principal problemática es que en Argentina no contamos con redes de este tipo. En nuestra ciudad (Salta) particularmente se presenta la problemática del crecimiento del parque automotor en este último tiempo, con lo cual las emisiones han aumentado y no se tiene ningún control sobre estas. Otro problema adicional es el costo que tienen estas estaciones. Al no contar con estas tecnologías no estamos siendo conscientes de la realidad ambiental en la que vivimos ni podemos realizar simulaciones para saber cómo podría ser la situación en unos años.

Objetivos

Lo que se espera de este proyecto de grado es brindar una herramienta que pueda ayudar a resolver la problemática de nuestra comunidad desde el punto de vista tecnológico, mediante el desarrollo de un sistema de monitoreo y análisis de al aire de bajo presupuesto. Para esto se empleara diferentes tecnologías de IoT (Internet de las Cosas) presentes en el mercado.

Se desarrollara un prototipo a escala para demostrar como funcionaria la solución propuesta.

Requisitos

Para referenciar correctamente los resultados obtenidos, se debe hacer uso del **ICA** (Índice de Calidad del Aire). Este índice es adoptado a partir de las recomendaciones técnicas que indica el *Technical Assistance Document for the Reporting of Daily Air Quality*[1]. El **ICA** perímete la comparación de los niveles de contaminación del aire en un tiempo *t*, que

[1] Manual de Operación de Sistemas de Vigilancia de Calidad de Aire del Protocolo para el Monitoreo y Seguimiento de la Calidad del Aire (MAVDT, 2010 Pág. 137 y 138).

corresponde al periodo de exposición previsto en la norma para cada uno de los contaminantes a medir.

Para cada contaminante se calcula un índice de referencia de manera particular. Para el proyecto se tendrá en cuenta solo el CO a fin de proveer una PoC (Proof of concept – Prueba de concepto).

Los valores del ICA son ubicados en 6 rangos o intervalos, los cuales componen una escala adimensional que va de 0 a 500. Cada rango o intervalo, simboliza la contaminación del aire y está relacionado directamente con la amenaza que representa para la salud humana. Para facilitar la lectura, se identifica a cada rango con un color de alerta.

RANGOS ICA	COLORES
0 ≤ ICA ≤ 50	**Verde**
51 ≤ ICA ≤ 100	**Amarrillo**
101 ≤ ICA ≤ 150	**Anaranjado**
151 ≤ ICA ≤ 200	**Rojo**
201 ≤ ICA ≤ 300	**Morado**
301 ≤ ICA ≤ 500	**Marrón** **(usualmente no es necesario mostrarlo)**

Tabla 2 - Rangos en los que han sido clasificados los valores del ICA de acuerdo

Color y calificación respecto a su efecto sobre la salud humana	Acciones preventivas
8h (ppm)	
VERDE (Buena)	Ninguna.
AMARILLO (Moderada)	
ANARANJADO **(Dañina a la salud para grupos sensibles)**	Niños activos, adultos y personas con enfermedades respiratorias como asma, deben reducir la actividad física al aire libre.
ROJO (Dañina a la salud)	Niños y adultos activos y personas con enfermedades respiratorias, deben reducir la actividad física prolongada al aire libre, especialmente los niños.
MORADO (Muy dañina a la salud)	Niños activos, adultos y personas con enfermedades respiratorias como asma, deben reducir la actividad física al aire libre; especialmente los niños
MARRÓN (Peligrosa)	Todas las personas deben evitar el esfuerzo al aire libre

Tabla 3 - Acciones preventivas dependiendo del valor del ICA para el

ICA	COLOR	CLASIFICACIÓN	O3 8h ppm	O3 1h ppm	PM_{10} 24h $\mu g/m^3$	$PM_{2.5}$ 24h $\mu g/m^3$	CO 8h ppm	SO_2 24h ppm	NO_2 1h ppm
0 - 50	Verde	Buena	0.000 0.059	-	0 54	0 12	0 4.4	0 0.035	0 0.053
51 - 100	Amarillo	Moderada	0.060 0.075	-	55 154	12.1 35.4	4.5 9.4	0.036 0.075	0.054 0.100
101 - 150	Naranja	Dañina a la salud para grupos sensibles	0.076 0.095	0.125 0.164	155 254	35.5 55.4	9.5 12.4	0.076 0.185	0.101 0.360
151 - 200	Rojo	Dañina a la salud	0.096 0.115	0.165 0.204	255 354	55.5 150.4	12.5 15.4	0.186 0.304	0.361 0.649
201 - 300	Púrpura	Muy Dañina a la salud	0.116 0.374	0.205 0.404	355 424	150.5 250.4	15.5 30.4	0.305 0.604	0.650 1.249
301 - 400	Marrón	Peligrosa	-	0.405 0.504	425 504	250.5 350.4	30.5 40.4	0.605 0.804	1.250 1.649
401 - 500	Marrón	Peligrosa	-	0.505 0.604	505 604	350.5 500.4	40.5 50.4	0.805 1.004	1.650 2.049

Tabla 4 - Punto de corte para el CO para el ICA

Para llevar a cabo el proyecto, también existen requisitos de hardware y software.

Requisitos de Hardware

Sera necesario contar con:

- Placas Arduino
- Placa Raspberry Pi
- Sensores CO y otros gases presentes en el aire.
- Sensores de presión y temperatura.
- Módulos LoRa
- Fuente de alimentación con voltaje variable
- Insumos de electrónica varios.

Requisitos de Software

El software necesario será:

- IDE Arduino
- Apache server
- Sistema operativo Windows/Linux
- IDE PHP STORM
- MySQL Server
- PUTTY
- POSTMAN
- GIT

Restricciones

- El sensor para detectar los niveles de CO a emplear, tiene una sensibilidad de 10-1000ppm, mientras según lo establecido por el **ICA,** a partir de los 9,5ppm puede ser dañino para la salud. Por lo que los valores inferiores no son 100 % precisos, sino más bien estimativos.
- El tiempo del proyecto tiene una duración máxima de 2 años.
- El tiempo en que se realizara el proyecto será de 11 meses (A excepción de las pruebas de campo).
- La distancia máxima entre nodos será de 4km en áreas pocas densas de construcciones y árboles. Queda como futura línea de investigación incrementar la distancia de los nodos.

Alcance

A continuación se detalla el alance del proyecto:

- Crear un sistema informático que sirva como herramienta para mejorar la calidad de vida de las personas en la ciudad. Cabe aclarar que dicho sistema informático es un prototipo funcional, el cual debería estar suficientemente perfeccionado y cumplir con normas y leyes, para recién implementarse de manera formal.
- La solución a implementar proporcionara el hardware y software de base necesario, como así también el equipamiento necesario para el sistema en la nube.

- Módulos del sistema en la nube:
  - Control de acceso
  - Gestión de nodos
  - Resultados de mediciones
  - Estadísticas y reportes en base a periodos de tiempos
  - Mapa para visualizar la ubicación de los nodos
- El sistema en la nube será accesible desde cualquier dispositivo que cuente con un navegador web.
- El sistema se desarrollara de forma modular para agilizar el desarrollo y mantenimiento del mismo.
- La conexión a internet es provista por un ISP externo, este servicio es ajeno al proyecto.
- El sistema podrá discriminar a los usuarios en base a roles y niveles de permisos, de esta forma cada usuario con diferentes roles tendrá restringido el acceso a ciertas funcionalidades y pantallas del sistema.
- Los parámetros a censar serán CO, presión atmosférica, temperatura y humedad.

Análisis FODA

El análisis FODA es una herramienta de análisis que tiene forma de matriz. Se aplica en cualquier situación, empresa o producto, que sea objeto de estudio en un determinado momento.

Figura 13 – FODA

Fuente: http://www.visionindustrial.com.mx -2019

En base a esto para la solución propuesta se plantea:

Fortalezas:

- Desarrollo de un producto de bajo presupuesto
- Gran oportunidad de escalabilidad del producto gracias a al hardware libre.
- Obtención de información en tiempo real.
- Puede ser de interés para entidades gubernamentales.

Oportunidades:

- Aumento de importación de insumos (más variedad de hardware y marcas).
- Campañas de concientización del medio ambiente.
- Buena aceptación y uso del sistema por parte de personas interesadas.

Debilidades:

- Bajo número de sensores incorporados.
- Disminución del alcance de transmisión en zonas muy urbanizadas.
- La topología de red estrella no es robusta.

Amenazas:

- Empresas que desarrollen una solución integral.
- Bloqueo de importaciones de productos para el armado del sistema.
- Temas inflacionarios por la situación económica del país.
- Modificaciones en leyes y normativas que afecten la transmisión y funcionamiento de los nodos.
- Fuertes tormentas que pongan en riesgo la integridad de los nodos.
- Hurto y/o destrucción de los nodos por parte de delincuentes.

Estrategias Ofensivas:

- Realizar impresión en masa del circuito para abaratar costos
- Difundir en escuelas y universidades el proyecto para incentivar a posibles colaboradores

Estrategias Adaptativas:

- Dar a conocer el proyecto a la sociedad y como pueden colaborar.
- Incentivar a las personas a participar mediante demostraciones y promoción en redes sociales.

Estrategias Reactivas:

- Abastecer de un stock considerable de hardware e insumos necesarios para evitar cualquier problema a futuro con las importaciones e inflación.
- Hacer un esfuerzo para minimizar el consumo de los nodos sensores y aprovechar las energías renovables.

- Invitar a expertos a dar consejos y opiniones para mejor el circuito, para que este sea más eficiente y disminuir costos.

Estrategias Defensivas:

- Implementar un canal de comunicación para dar soporte a colaboradores y mejorar gradualmente el sistema.
- Generar guías para construir nodos con hardware alternativo, de esta manera se incrementaría el número de posibles colaboradores del proyecto.
- Continuar con la investigación para analizar y probar sensores que briden mejor resultado y determinar cual tiene la mejor relación costo/beneficio.

4-Solución propuesta

La propuesta central del presente proyecto de grado fue controlar la calidad del aire en áreas metropolitanas para obtener datos y analizar los mismos para poder tomar decisiones que permitan mejorar la calidad de vida de las personas y cuidado del medio ambiente.

Para poder lograr este objetivo me enfoque en los parámetros que definen la calidad del aire según la Organización Mundial de la Salud, siendo más preciso, en el análisis de concentración de Monóxido de Carbono en el aire. La solución propuesta desde el punto de vista de la informática, consiste en permitir recaudar los datos y presentarlos mediante gráficos para una mejor comprensión. Empleando hardware asequible y de bajo presupuesto.

Se ideo una red de nodos los cuales comparten la información con un nodo Gateway que es el encargado de subir los datos a la nube para ser almacenados, analizados y presentados. El flujo del sistema será el siguiente:

1. Los nodos Sensores (que cuentan con el hardware necesario para hacer las mediciones) realizaran mediciones en intervalos de tiempo de 2 minutos.
2. Cada 2 minutos los datos obtenidos son enviados por los nodos Sensores al nodo Gateway, el cual es el único nodo que se encuentra con conexión a internet.
3. El nodo Gateway tras recibir los datos, se encarga de subirlos a un servidor en la nube mediante una api desarrollada en PHP.
4. El servidor en la nube recibe la información, guarda en una base de datos los datos.
5. El usuario puede acceder a los datos almacenados en la base de datos mediante un sistema el cual permite hacer consultas cronológicas de los datos y generar gráficos para realizar análisis y comparaciones.

Análisis de Factibilidad

Factibilidad Económica

El software empleado es gratuito, salvo el IDE de PHPStorm el cual es de pago pero permite el uso de licencia estudiantil sin costo alguno.

Los costos de hardware e insumos necesarios para la construcción básica se muestran a continuación:

Hardware e Insumos	Precio	Cantidad	Total	Fecha de presupuesto
Arduino mini pro 5V		1	USD 4,19	25/10/2019
Sensor MQ-9		1	USD 3,06	25/10/2019
Sensor DHT11		1	USD 1,93	25/10/2019
Módulo transceptor sx1278		2	USD 22,55	25/10/2019
Lm7805 7805 Regulador Tensión Voltaje 5v		1	USD 1,13	25/10/2019
Módulo Alimentación 5v A 3v3 800ma Ams1117 ó regulador LM		1	USD 0,80	25/10/2019
Fuente para protoboard		1	USD 2,25	25/10/2019
Módulo Sensor BMP280		1	USD 3,87	25/10/2019
Tarjeta Micro SD 8gb clase 10		1	USD 4,83	25/10/2019
Cargador 5V		2	USD 6,44	25/10/2019
Raspberry pi Zero		1	USD 38,65	25/10/2019
Estaño Eximetal para soldar 100gr		1	USD 6,44	25/10/2019
Batería 9V		1	USD 2,42	25/10/2019

Conector para batería de 9V	1	USD 0,52	25/10/2019
Plaqueta perforada	1	USD 1,21	25/10/2019
Cables para protoboard	pack	USD 4,83	25/10/2019
Tira de pines machos	5	USD 1,79	25/10/2019
Tiras de pines hembra	5	USD 1,79	25/10/2019
Protoboard	1	USD 2,90	25/10/2019
Soldador	1	USD 6,44	25/10/2019
Caja estanco 10x10cm	1	USD 1,93	25/10/2019
		Total	
		USD 88,24	

Tabla 5 - Costos de Hardware

En cuanto a los gastos de RR.HH fueron estimados según lo dispone el COPAIPA (Consejo profesional de Agrimensores, Ingenieros y Profesiones Afines) de la Provincia de Salta al día 08/03/2019:

Profesional	Honorarios por hora	Horas de trabajo diarias	Días de trabajo	Costo Total
Analista Funcional	USD 5,50	4	60	USD 330

Tabla 6 -Costos de RRHH

También es necesario para que se pueda desarrollar el sistema que adquiera una computadora:

Equipo	Precio Unitario	Cantidad	Total	Fecha de Presupuesto
PC Intel Core I7 8550u 8gb 1tb 15.6 8°gen	$30.000	1	USD 483	25/10/2019

Tabla 7 - Costos de Equipamiento

Por último se requiere contratar un servidor para alojar el aplicativo web y la base de datos, además de registrar un nombre de dominio.

Servidor	Costo anual	Fecha de Presupuesto
Digital Ocean 1vCPU 3GB RAM 250GB	USD 618	25/10/2019

Tabla 8 - Costos de Servidor

Proveedor de DNS	Costo Anual	Fecha de Presupuesto
Hostinger	USD 10	25/10/2019

Tabla 9 - Costos de DNS

Tipo de Costo	Subtotal
Hardware e Insumos	USD 88,24
RR.HH	USD 330
Equipamiento	USD 483
Servidor	USD 618
Nombre de Dominio	USD 10
	TOTAL: USD 1529.24

Tabla 10 - Total de costos del proyecto

Aclaración: El costo de hardware es para un nodo sensor y un nodo Gateway, por lo que se detalla a continuación el precio unitario para cada uno:

TIPO DE NODO	COSTO POR UNIDAD
NODO SENSOR	**USD 36**
NODO GATEWAY	**USD 45**

Tabla 11 - Costos por Nodos

Beneficio Económico Esperado

El Proyecto en sí no tiene ningún fin lucrativo por lo que no es posible cuantificar en términos económicos los beneficios generados, no al menos de una manera simple. Es un proyecto que aporta al medio ambiente, la sociedad y la calidad de vida de la misma.

"Planta un árbol y estarás sembrando conciencia" Desconocido

Al tratarse de un proyecto sin fines de lucro se podría ver financiado por entidades gubernamentales o diferentes fuentes alternativas como fundaciones y movimientos sociales.

Factibilidad Técnica

Es posible el desarrollo del sistema empleando cualquier PC de gama media/alta de la actualidad ya que no se requiere demasiado procesamiento ni algoritmos complejos. Como es de esperarse el servidor si deberá contar con un buen rendimiento, sobre todo a medida que se expanda la red y se incorporen más nodos sensores que alimenten la base de datos.

Todos los insumos de hardware detallados para el armado del sistema son adquiribles, si bien no son productos de fabricación nacional en su mayoría, se pueden conseguir sin problemas (de momento).

Factibilidad legal

En la República Argentina existe un ente encargado de gestionar y administrar los espectros de radio frecuencia. Este ente es ENACOM. La frecuencia a operar es de 433.3Mhz y es de libre uso en el país.

Gestión de Riesgos

Un riesgo, que es un problema potencial que puede ocurrir o no. Siempre que se quiera desarrollar un proyecto es recomendado analizar los riesgos que podrían presentarse como así también la mitigación o plan de acción a efectuar en caso de que se presente alguno.

En la siguiente tabla se detallan los riesgos asociados al proyecto, acción para mitigarlos, la probabilidad de que suceda y el impacto que podría llegar a producir el mismo:

Riesgo	Acción para Mitigar	Probabilidad de suceso	Impacto
Robo o destrucción de un nodo	Dificultar el acceso al mismo. Ej.: colocarlos a una altura considerable del suelo. Elaborar una carcasa resistente para proteger los circuitos internos.	50%	medio
Destrucción de un nodo por mal tiempo	Prueba de mejoras en cuanto a durabilidad y resistencia de los materiales	40%	medio

	empleados para la construcción del nodo.		
Incremento de los costes de hardware por inflación	Adquirir productos por grandes cantidades para abaratar costos.	%70	alto
Cambios en las normativas /leyes referentes a las transmisiones de radio frecuencia	Migrar a otra frecuencia diferente para transmitir los datos. Analizar si existe la posibilidad de seguir haciéndolo, cumpliendo determinados requisitos (si es que existe la posibilidad)	20%	bajo
Pérdida de calidad en la comunicación	Incrementar el número de gateways en la zona. Disminuir la distancia entre los nodos	50%	Medio-Alto

Tabla 12 - Tabla de Riesgos

Análisis comparativo de conectividad

Hoy en día existen diversos protocolos y tecnologías que hacen posible la comunicación entre dos o más dispositivos. En grandes rasgos podemos categorizarlos en base al alcance, consumo energético, dificultad de implementación, si se tratan de uso libre o no, entre otras características. Para este proyecto es importante seleccionar un medio de comunicación que cuente con bajo consumo energético y un alcance de distancia media (en km).

Como se mencionó con anterioridad existen diversas tecnologías que hacen posibles la comunicación y se seleccionaron las más prometedoras para aplicar en este proyecto en particular, realizando un análisis comparativo entre las mismas.

	LTE-M	NB-IoT	LoRa	SIGFOX
BAJO CONSUMO	PARCIAL	SI	PARCIAL	SI
ANCHO DE BANDA	SI	SI	SI	PARCIAL
ESTÁNDAR ABIERTO	PARCIAL	PARCIAL		PARCIAL
DESARROLLO		PARCIAL	SI	SI
SEGURIDAD	SI	PARCIAL	PARCIAL	PARCIAL

Tabla 13 - Comparativa de Tecnologías

Para este proyecto se empleó LoRa como protocolo de conexión entre los nodos.

¿Qué es LoRa y porque se usó?

Figura 14 – LoRa

Fuente: https://www.semtech.com/lora

LoRa es una tecnología de comunicación que permite el envío y recepción de información punto-a-punto. La principal característica de estos dispositivos es su pequeño tamaño y consumo de energía. Se denominan dispositivos Long Range, en español de largo alcance, porque en situaciones ideales pueden llegar a más de 20km de distancia.

Para lograr un largo alcance emplea una técnica de modulación para la transición de datos digitales por radiofrecuencia llamada espectro expandido o disperso (spread spectrum o ss.). Con esta técnica, la señal a enviar utiliza mucho más ancho de banda del el necesario teóricamente, pero que permite una recepción de múltiples señales a la vez que tengan distinta velocidad. El ancho de banda transmitido se determina mediante alguna función independiente del mensaje y que es conocida por el receptor.

La empresa encargada de crear LoRa y de su patenta miento es **Semtech**, una empresa multinacional que cuenta con más de 30 sedes alrededor del mundo. En la actualidad existe una fundación llamada **LoRa Alliance** que se encarga del desarrollo del estándar y de su evolución.

Se pueden usar 3 frecuencias que son 433 Mhz, 868 Mhz y 915 Mhz. Estas frecuencias pueden estar restringidas según el país. Para este proyecto se optó por la frecuencia de 433Mhz ya que es una frecuencia libre para radios aficionados en Argentina.

REGIÓN 2 – UIT	REPÚBLICA ARGENTINA
420 - 430 **(Cont.)** FIJO MÓVIL Salvo móvil aeronáutico Radiolocalización 5.269 5.270 5.271	**426,0125 - 426,2375** ATRIBUCIÓN CONDICIONADA N11
	426,2375 - 426,5 FIJO MÓVIL salvo móvil aeronáutico
	426,5 - 430 FIJO MOVIL
430 - 432 RADIOLOCALIZACIÓN Aficionados 5.271 5.276 5.278 5.279	**430 - 432** AFICIONADOS
432 - 438 RADIOLOCALIZACIÓN Aficionados Exploración de la Tierra por satélite (activo) 5.279A 5.271 5.276 5.278 5.279 5.281 5.282	**432 - 433,075** AFICIONADOS
	433,075 - 434,775 AFICIONADOS Móvil
438 - 440 RADIOLOCALIZACIÓN Aficionados 5.271 5.276 5.278 5.279	**434,775 - 440** AFICIONADOS Aficionados por satélite
440 - 450 FIJO MÓVIL Salvo móvil aeronáutico Radiolocalización 5.269 5.270 5.271 5.284 5.285 5.286	**440 - 443** FIJO MOVIL

Figura 15 - cuadro de atribución de bandas de frecuencias de la República Argentina – Enacom

¿Qué es LoraWAN?

Es un tipo de red LPWAN (Low Power Wide Area Network). Dentro del modelo OSI, correspondería al nivel 2, es decir la capa MAC (Media Access Control). LoRaWAN es quien se encarga de conectar diferentes dispositivos LoRa gestionando los canales y parámetros de conexión, ancho de banda, cifrado de datos, etc.

Procedimiento lógico del Sistema

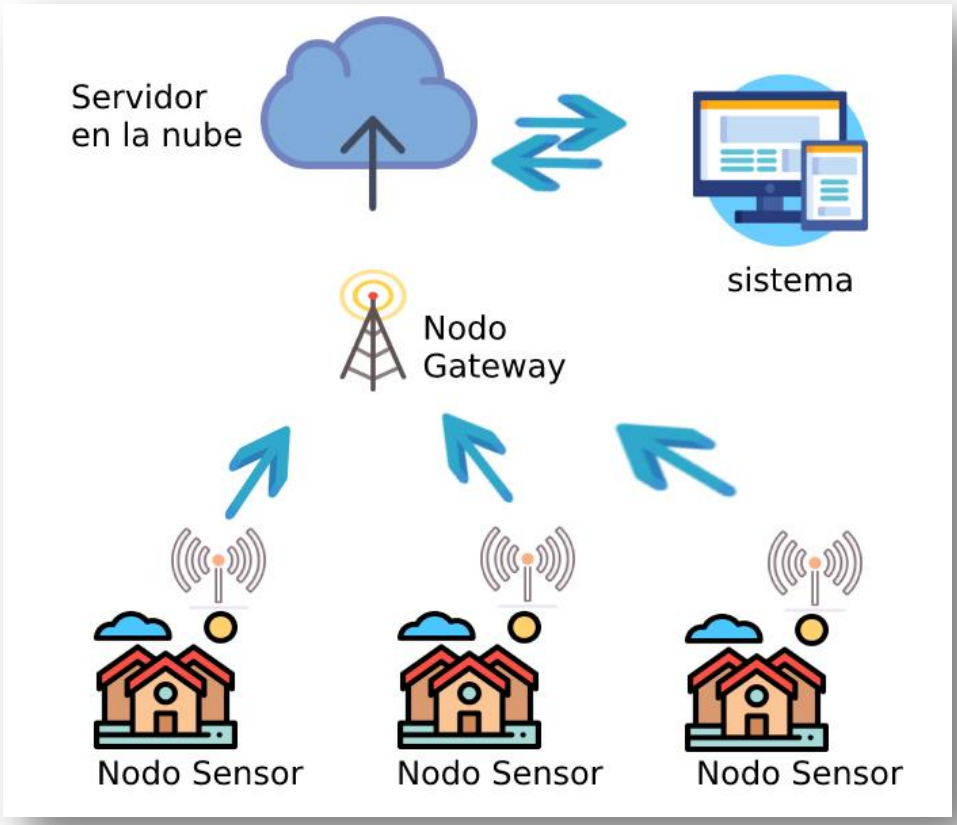


Figura 16 - Diagrama simplificado

Antes de detallar el funcionamiento del sistema es necesario comprender que este se compone en tres componentes principales que son:

1. **Los nodos Gateway:** ejecuta un script en Python que se encarga de poner el transceptor LoRa a la escucha. Dentro del nodo sensor existe una lista blanca, la cual contiene los ID de los nodos sensores que pueden enviar datos. Esta lista blanca es para evitar escuchar los paquetes de otros nodos sensores que pudieran ser colocados en la zona y que estén trasmitiendo en esa frecuencia.

Cuando el nodo Gateway recibe un nuevo paquete, verifica que no contenga errores, verifica que el ID del nodo Sensor que lo envió este en la lista blanca y lo envía al servidor en la nube, mediante una conexión WIFI.

2. **Los nodos Sensores:** Se encarga de medir la temperatura en grados °C, el porcentaje de humedad, cantidad de CO (monóxido de carbono) en ppm (partes por millón). Las mediciones son tomadas cada 2 minutos y trasmitidas por radio frecuencia al nodo Gateway mediante el transceptor LoRa. Junto con los datos de las mediciones se envían la fecha y hora en que se realizaron las mediciones, junto con un ID que identifica el nodo sensor que envió los datos.

3. **El servidor**: recibe la información y verifica si existe algún registro en la base de datos que contenga el ID del nodo sensor que envió el paquete, de no existir ninguno, entonces se trata de un nuevo nodo que fue añadido a la red, con lo cual procede a registrarlo en la base de datos. Si el ID ya existe, entonces carga los datos de ese nodo.

Una vez que los datos son almacenados, estos pueden ser consultados mediante la interfaz web.

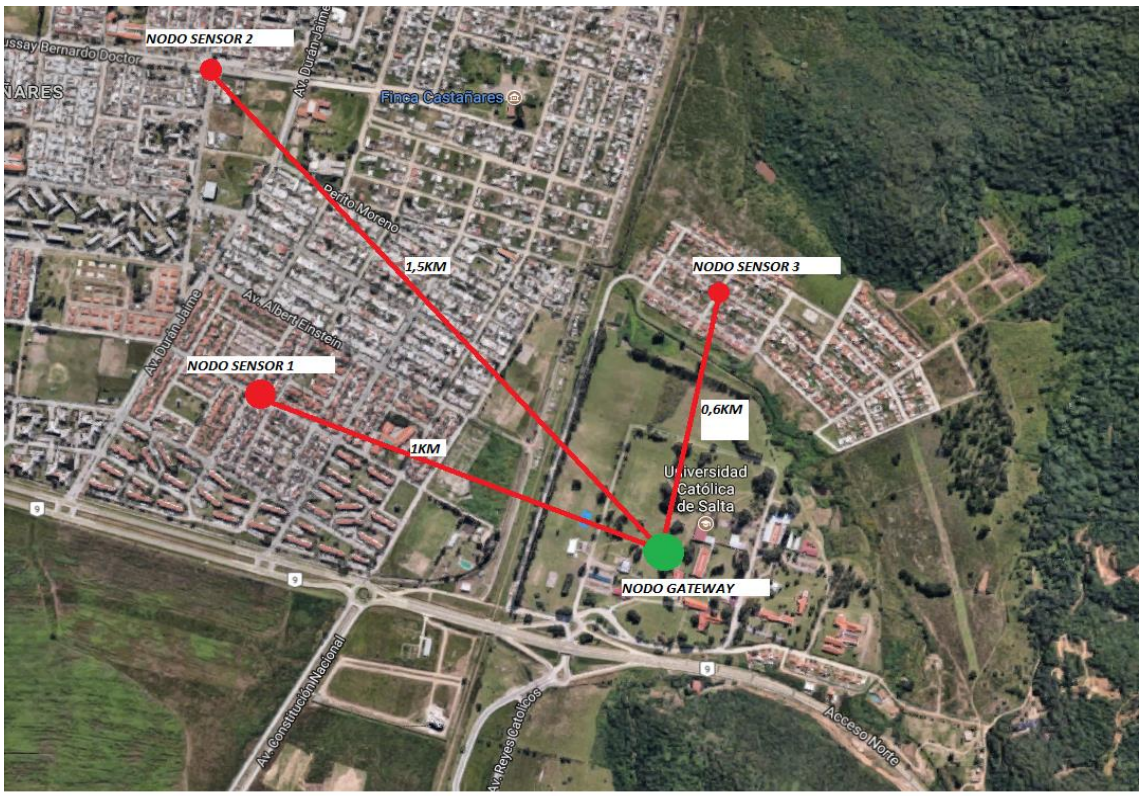


Figura 17 - Ejemplo ilustrativo de topología de red

Hardware

En lo que respecta al hardware empleado se divide en base a dos tipos de nodos, los nodos Gateway y los nodos sensores.

Nodos Gateway

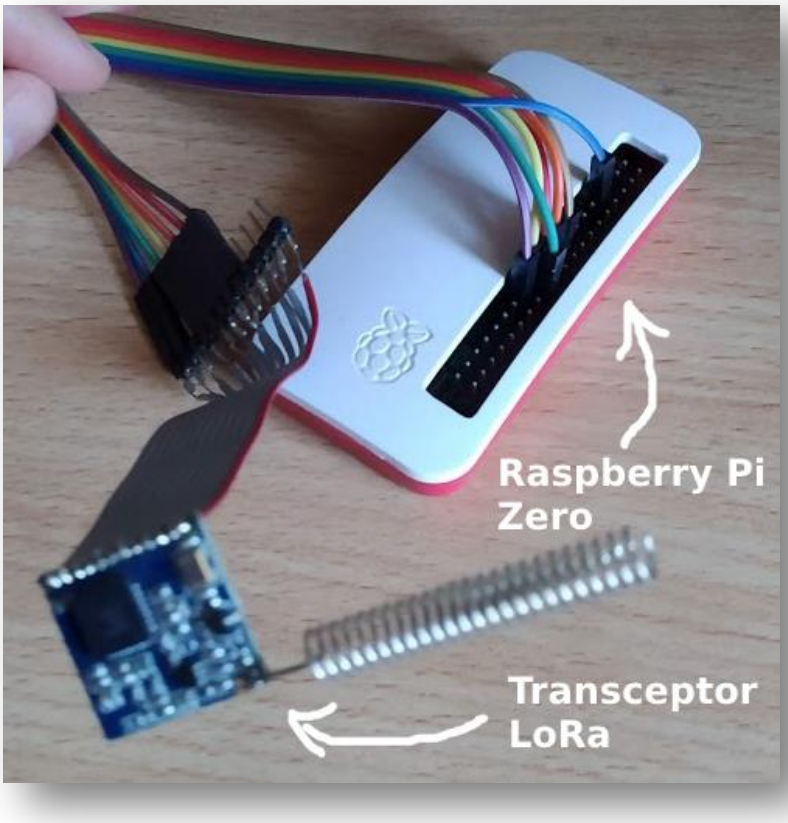


Figura 18 - Nodo Gateway

El hardware que compone un nodo Gateway está dado por:

- **1 Raspberry Pi Zero: Mini computadora de bajo costo que cuenta con:**
  - Micro procesador 1GHz ARM11 core
  - 512MB de LPDDR2 SDRAM
  - Una ranura micro-SD
  - una mini-HDMI entrada para salida de video 1080p60
  - Puertos Micro-USB para alimentación y datos
  - 40-pin GPIO

- **1 Modulo Transceptor Lora1278 Sx1278 Rf 4km**
    - Rango de Frecuencia: 433 / 470MHz (adaptable 137 a 1.020 MHz)
    - Sensibilidad -139 dBm hasta
    - Potencia máxima de salida: 20 dBm
    - velocidad de transferencia de datos: @ FSK, 1.2-300kbps
    - Lora TM, FSK, GFSK y el modo de modulación OOK
    - suministro de 1.8 a 3.7 V Energía
    - 127 dB Rango dinámico RSSI
    - equipo de paquete de hasta 256 bytes con CCR
    - Rango de temperatura: -40 ~ + 85 ° C
- **1 Fuente 5V 2.5ª**

A continuación se listan los planos de circuito:

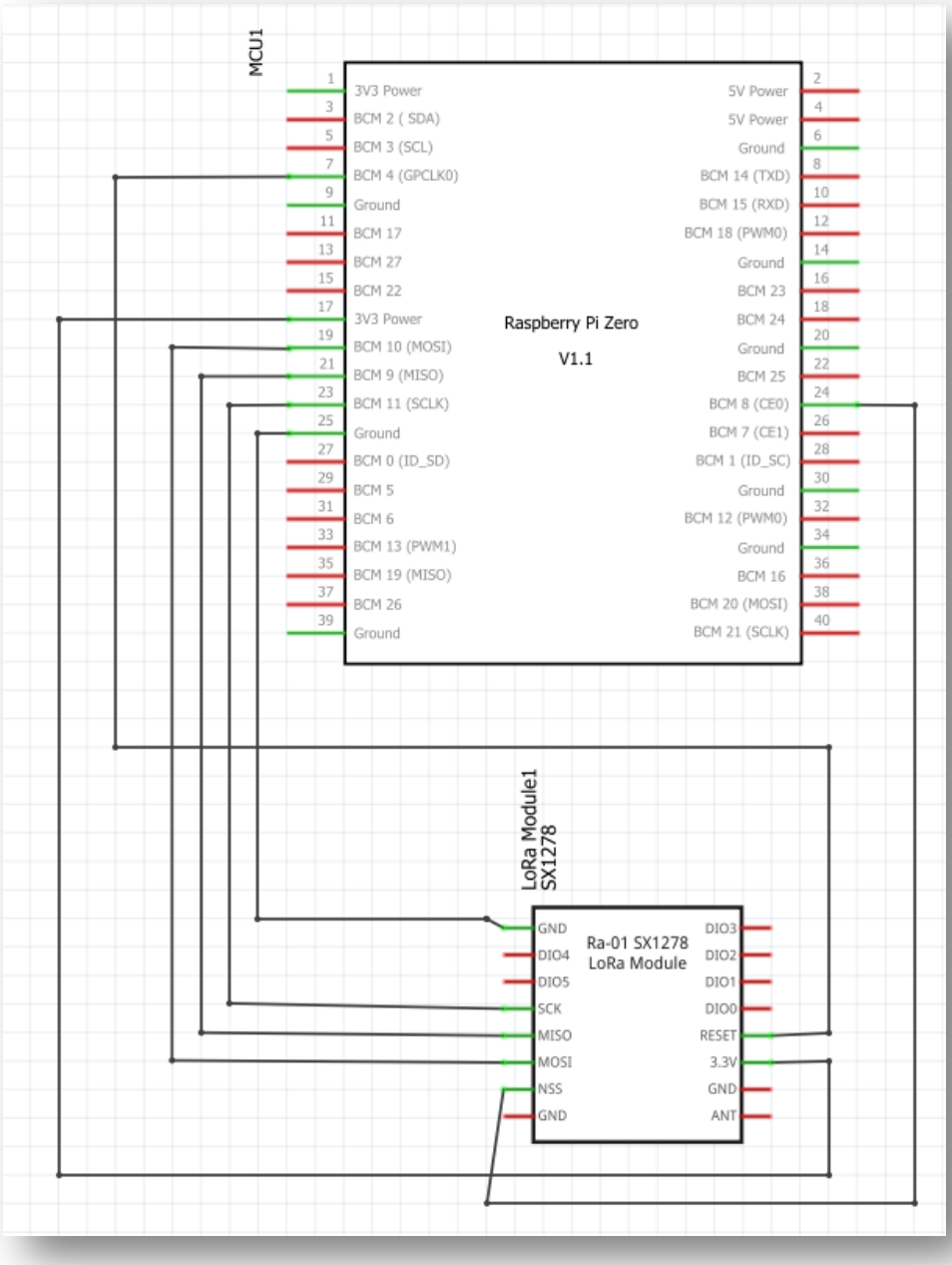


Figura 19 - Diagrama de circuito nodo Gateway

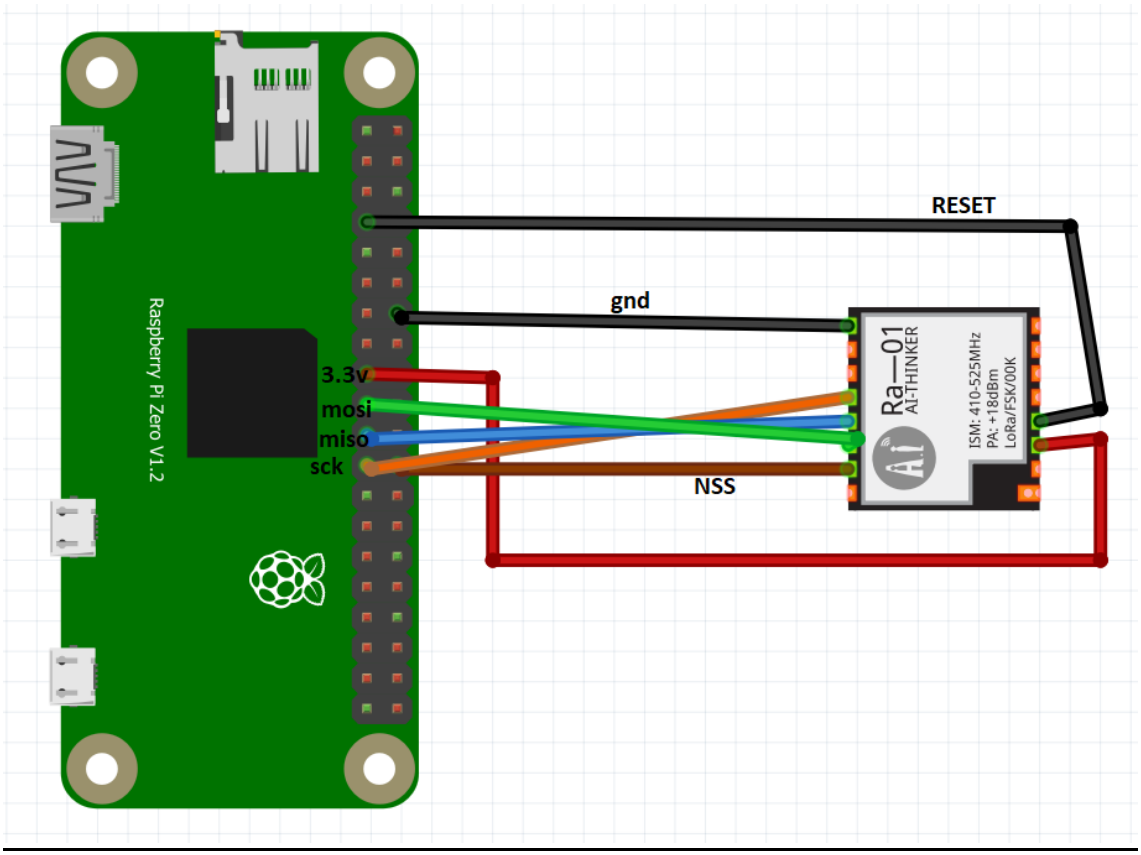


Figura 20 - circuito con componentes ilustrados - nodo Gateway

Especificación de conexión de pines para el nodo Gateway:

RASPBERRY PI ZERO	MODULO LoRa SX1278
GND pin 25	**GND**
3V3 pin 17	**3.3V**
GPIO4 pin 7	**RST (Reset)**
CS/CE0 pin 24	**NSS (CS chip select in)**
SCK pin 23	**CSK (SPI clock in)**
MOSI pin 19	**MOSI (SPI Data in)**
MISO pin 21	**MISO (SPI Data out)**

Tabla 14 - Conexiones modulo LoRa a Raspberry pi

Nodos Sensores

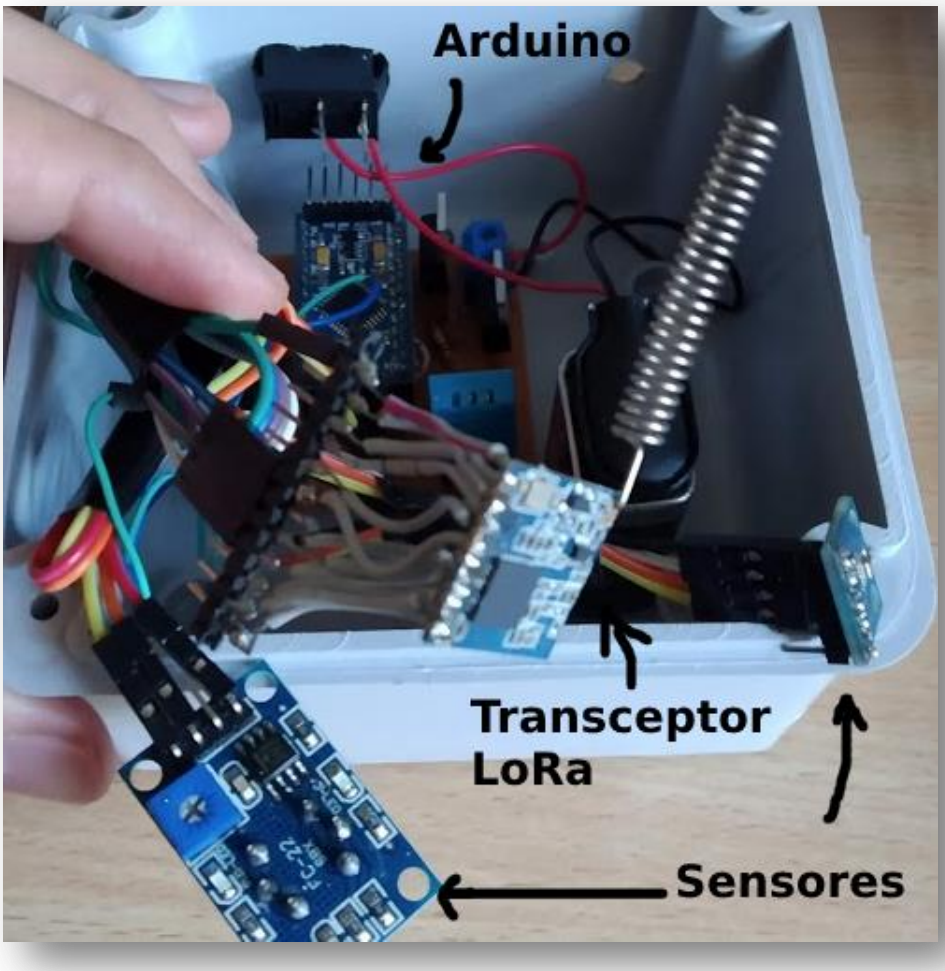


Figura 21 - Nodo Sensor

El hardware que compone un nodo sensor está dado por:

- *1 Arduino Mini PRO*
- *1 Modulo Transceptor Lora1278 Sx1278 Rf 4km*
- *1 Sensor MQ-9 de gas*
- *1 Sensor de presión y temperatura BMP280*
- *1 Sensor de Humedad DHT11*
- *1 Bornera*

- *1 Regulador de voltaje 12v – 5v (opcional)*
- *1 Regulador de voltaje 5v – 3.3 v*
- *1 Resistencia 10k*
- *1 Placa perforada 5x5 cm*
- *1 Batería 9v o 4 pilas AAA*
- *Un switch*
- *1 Caja estanco 11x11x8 cm.*

A continuación se detallan los planos del circuito para comprender como se realizan las conexiones:

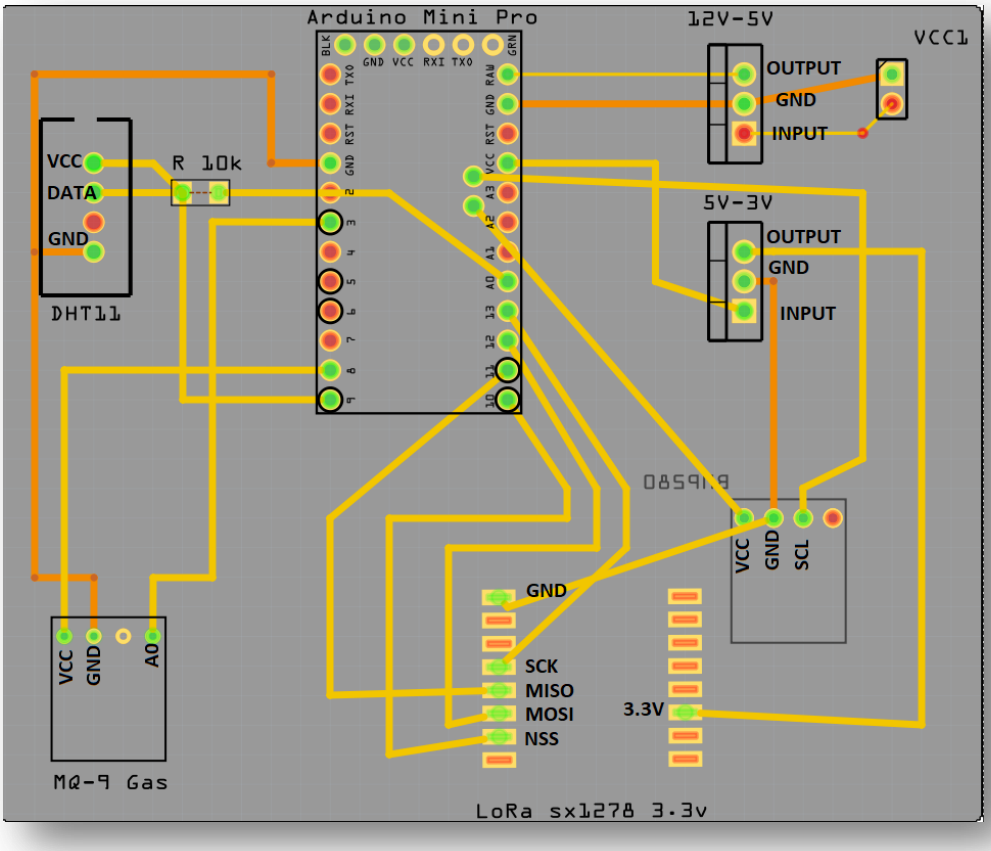


Figura 22 - PCB Circuito nodo Sensor

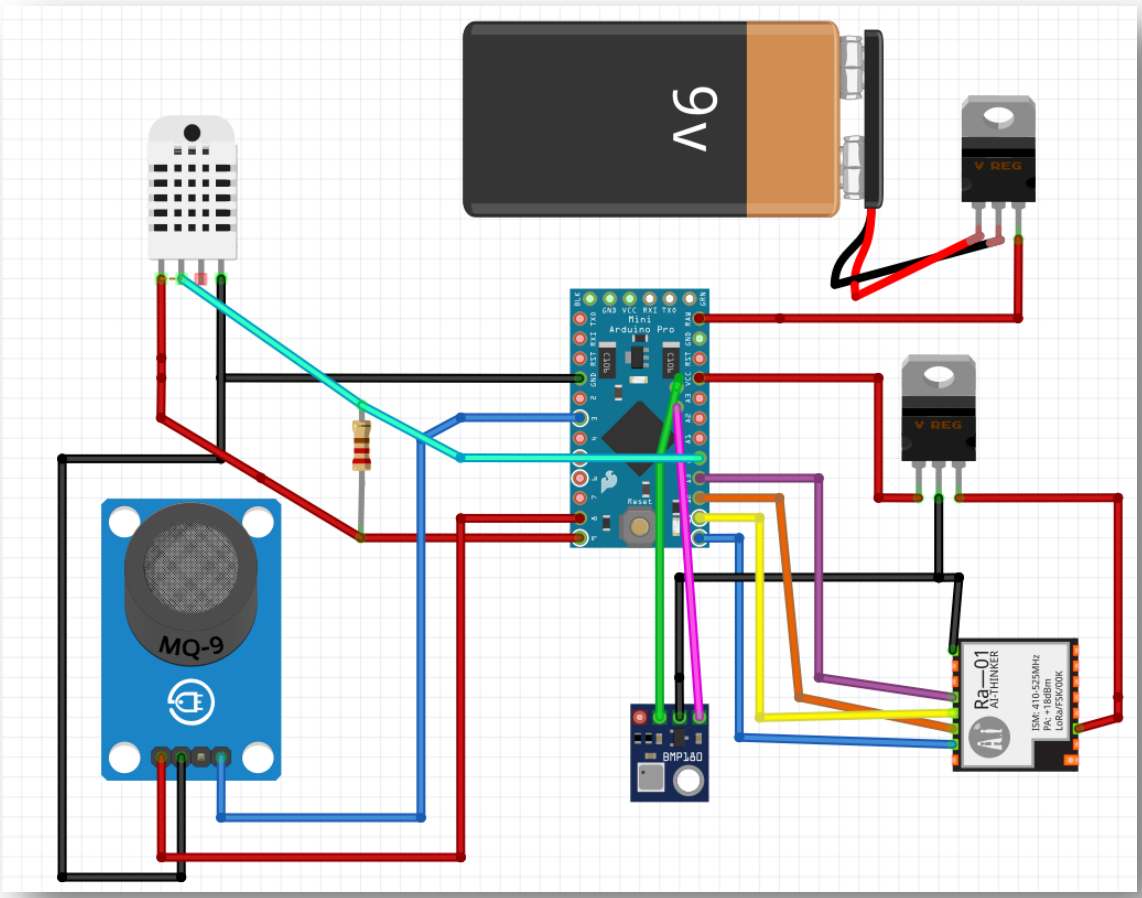


Figura 23 - Esquema con componentes ilustrados - nodo Sensor

Cabe destacar que el modulo LoRa que figura en los diagramas de circuitos difiere al empleado, ya que son diferentes fabricantes, pero la conexión y configuración de ambos es la misma.

ARDUINO MINI PRO	MODULO LoRa SX1278
PIN GND	**GND**
PIN 13	**SCK**
PIN 11	**MOSI**
PIN 12	**MISO**
PIN 10	**NSS**
SALIDA 3.3V REGULADOR	**VCC**

Tabla 15 - Conexión Arduino Modulo LoRa

Software

EL software fue desarrollado sobre una plataforma web basada en el framework de PHP Laravel, para la visualización y análisis de los datos, mientras que la obtención de los datos se realizó mediante un servicio REST también desarrollado con este framework. La programación del micro controlador Arduino que controla los sensores, se realizó en el lenguaje de programación que provee el mismo, el cual está basado en C++. En cuanto al software de lado del nodo Gateway, el cual hace uso de una Raspberry Pi Zero, se desarrolló un script en Python en conjunto con un framework para el desarrollo de nodos LoRa.

Se utilizó el IDE PHP Storm y Arduino IDE para su codificación. Como así también otras herramientas como PUTTY y POSTMAN.

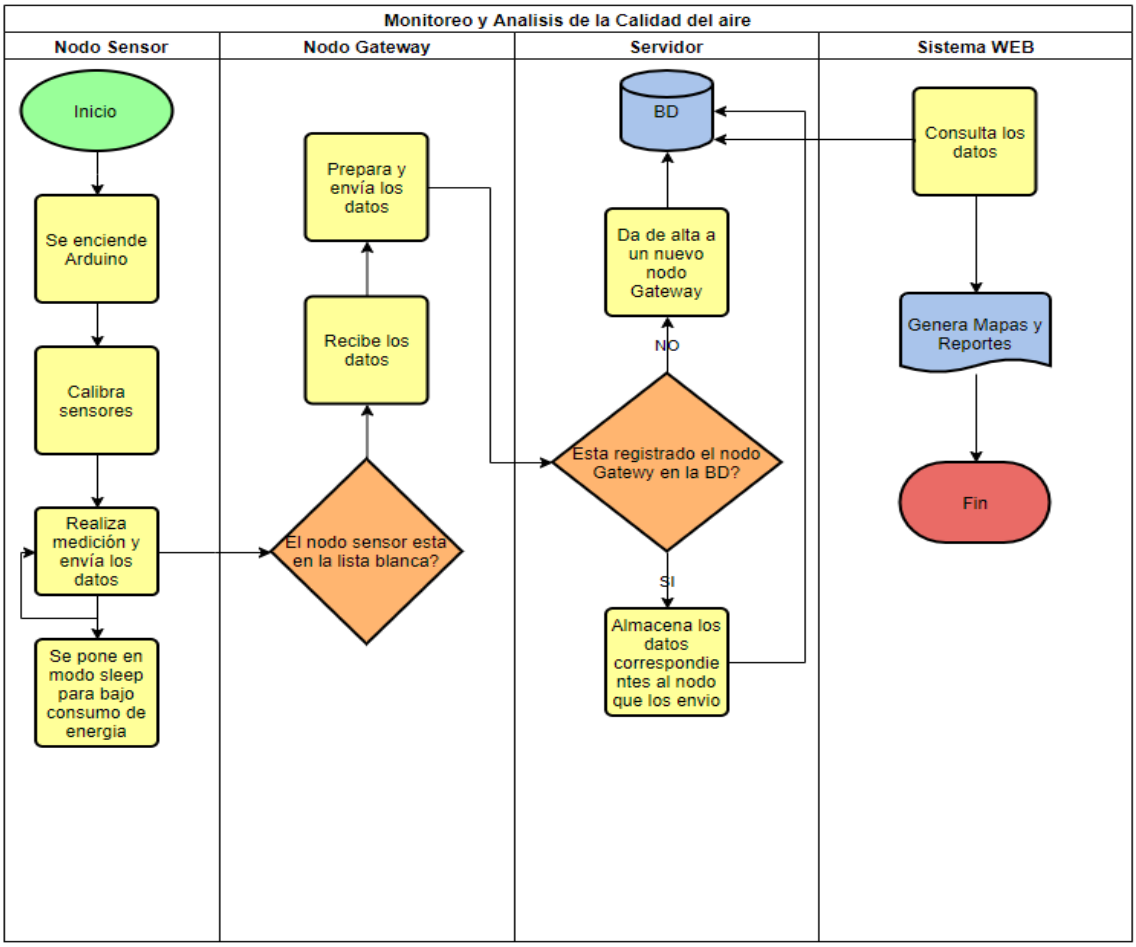


Figura 24 - Diagrama de flujo del sistema

El sistema posee los siguientes módulos:

- Módulo de control de acceso: Permite restringir el acceso al área de configuración del sistema.
- Módulo de reportes: Es el modulo encargado de generar reportes a partir de los datos almacenados en la base de datos.
- Módulo de nodos: Se encarga de mostrar los gateways y sensores existentes conectados al sistema, como así también ver las últimas actividades de estos y la ubicación de estos en un mapa.
- Módulo de comunicación: Este módulo es el encargado de controlar la comunicación entre el nodo gateway y el servidor donde reside el sistema de reportes.

El sistema se basa en MVC (Modelo – Vista – Controlador) donde el backend es un API REST y el frontend consume dicha API REST.

Los módulos ya mencionados se detallan incluyendo vistas para una mejor comprensión del funcionamiento del sistema.

Módulo de control de acceso

Para realizar cualquier tipo de modificación o eliminación de los datos registrados por los diversos nodos, se requiere ser un usuario autorizado con privilegios de administrador. Para ello se dispone de una pantalla de acceso en la cual se debe informar el nombre de usuario y la contraseña.

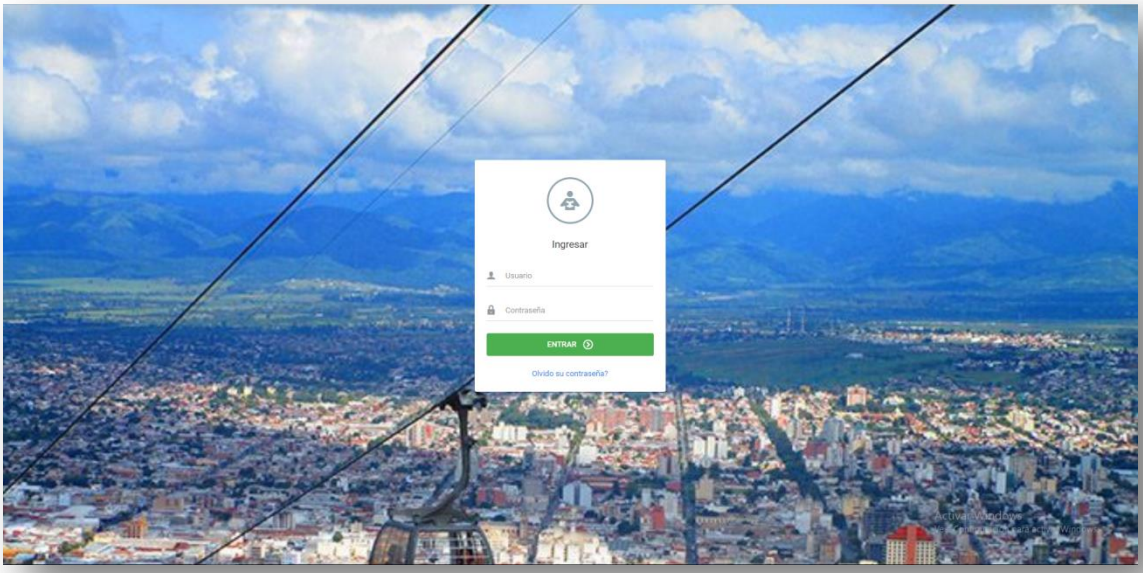


Figura 25 - Pantalla de Login

En cuanto a cualquier modificación que se requiera hacer de un nodo Gateway, se deberá estar conectado en la misma red WIFI y acceder mediante protocolo SSH informado usuario y contraseña.

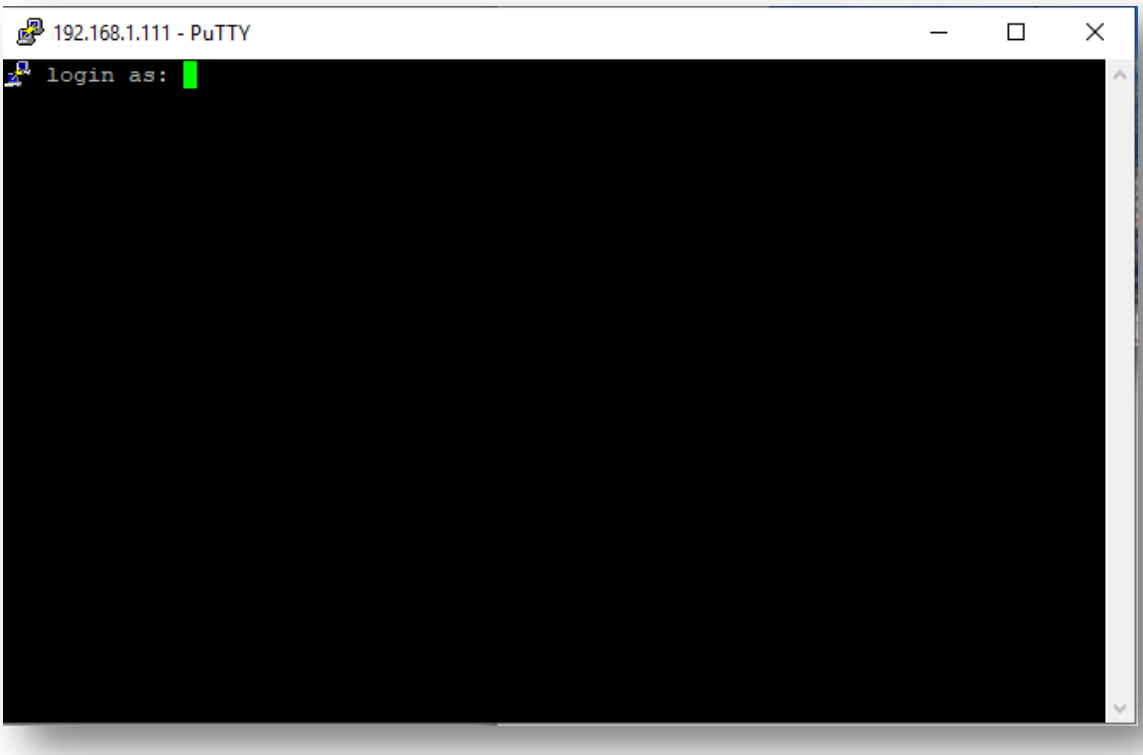


Figura 26 - Conexión remota mediante SSH a un nodo Gateway

Módulo de Reportes

Para acceder a los reportes no es necesario contar con privilegios de administrador, a menos que se desee modificar algún dato referente al nodo. En la pantalla de detalle de cada nodo Sensor se pueden observar gráficos que indican los niveles de CO actual, el promedio en las últimas 8 horas y un gráfico que indica la fluctuación de los mismos durante las últimas 24 horas y durante la última semana. También se indica la posición geográfica del nodo Sensor mediante el uso de un mapa.

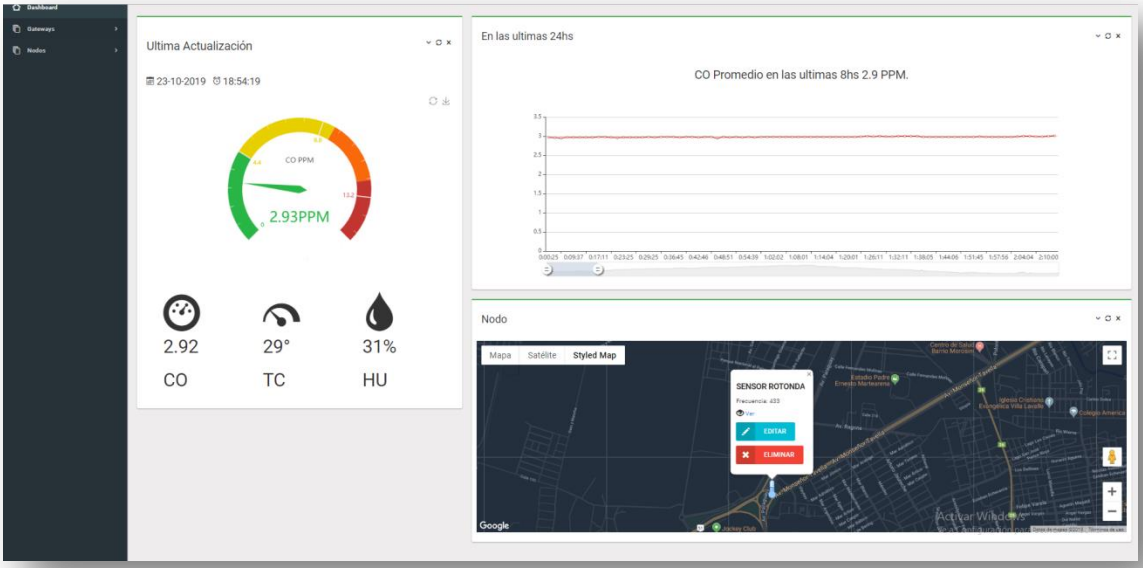


Figura 27 - Pantalla de detalle nodo Sensor Rotonda

En esta pantalla, los datos están constantemente actualizándose y modificando las gráficas en tiempo real. Se muestran los últimos datos obtenidos, el nivel promedio de CO en las últimas horas, un mapa con información geo posicional del nodo y dos gráficos de la última semana con datos de temperatura y niveles de CO.

Figura 28 - Pantalla de detalle del nodo sensor

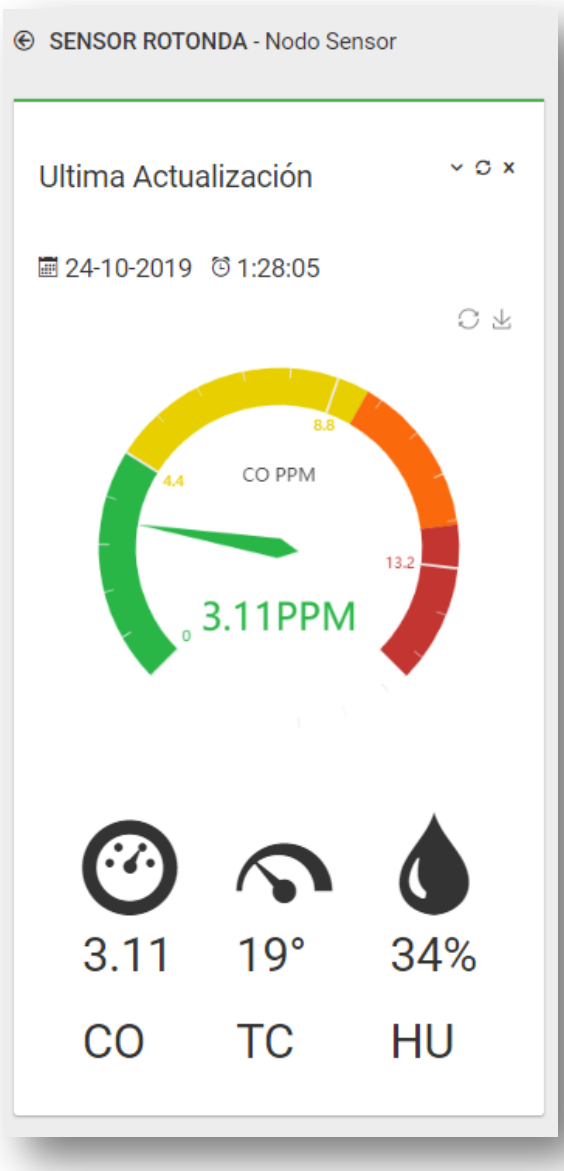


Figura 29 - Últimos datos registrados por el sensor

Módulo de Nodos

En este módulo se detallan en un mapa, todos los nodos existentes en el sistema, tanto sensores como gateways, dando a conocer información de los mismo y permitir la modificación o eliminación de estos.

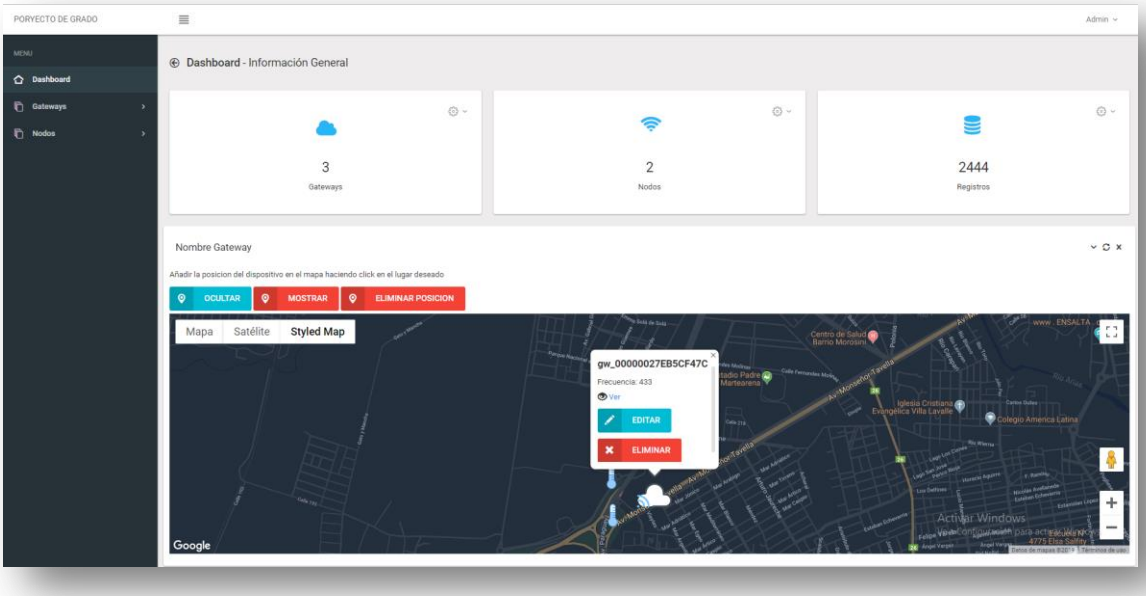


Figura 30 - Pantalla dashboard (menú principal)

En la pantalla principal, denominada dashboard se puede apreciar un mapa donde se indica la posición geográfica de cada nodo y se los diferencia mediante iconos a los nodos gateways de los nodos sensores. Al hacer click en un icono, se muestra una pequeña ventana con información reducida del nodo y opciones para su administración. También en la parte superior de la pantalla se listan la cantidad de nodos gateways, sensores y registros que existen actualmente en el sistema.

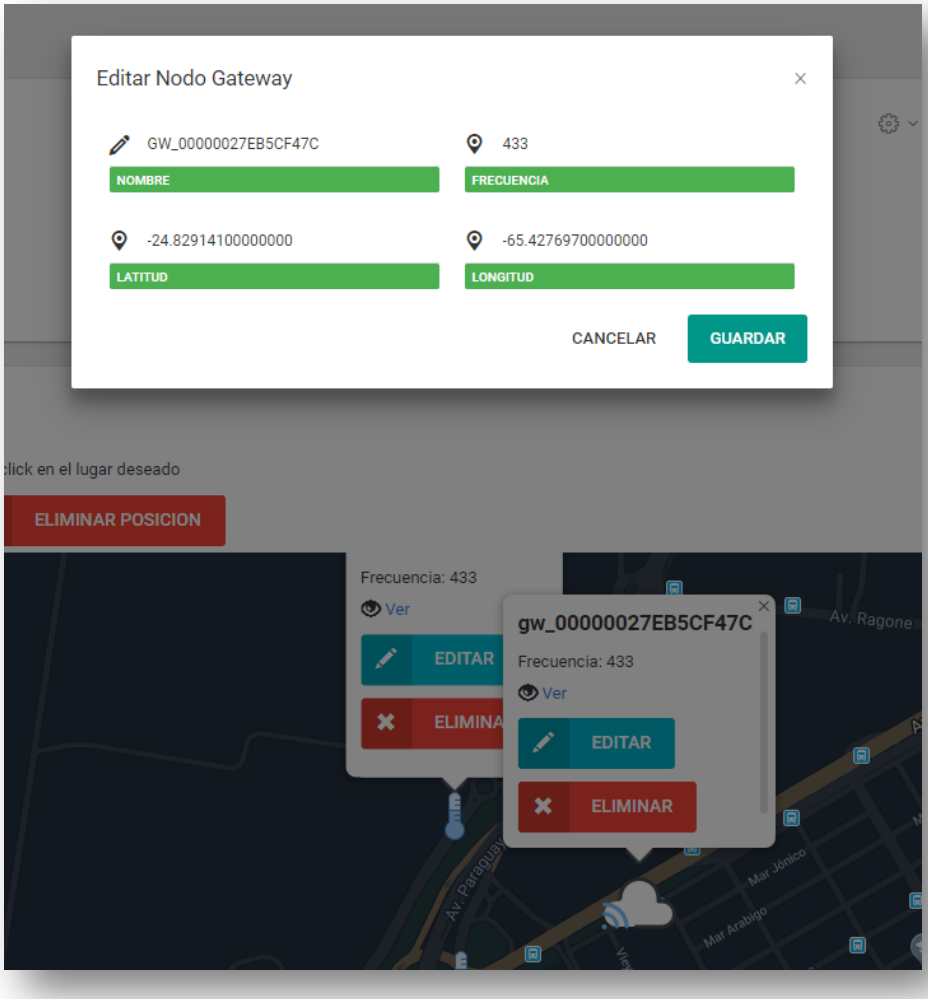


Figura 31 - Formulario emergente para editar datos del nodo

Para facilitar la busqueda y administracion de los nodos, se dispone de una pantalla en la cual se listan los mismos.

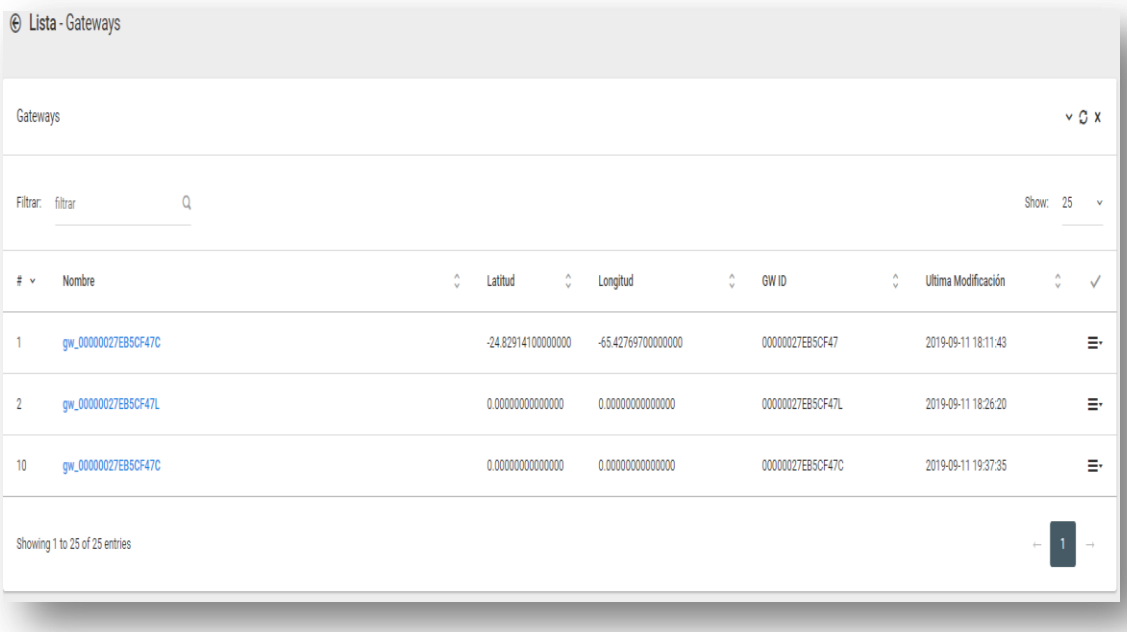


Figura 32 - Lista de nodos Gateways

Módulo de comunicación

Como ya se explicó anteriormente, el nodo sensor es quien se comunica con el nodo gateways y este último con el servidor donde reside el sistema, por ende, se divide el módulo de comunicación en dos partes.

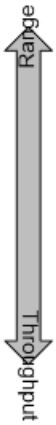
Comunicación desde el nodo sensor

Se debe configurar la placa arduino con el software que permite realizar las mediciones y enviar la información. Son necesarios diversos parámetros para que este funcione correctamente, pero entre los más importantes que encontramos, son aquellos que deben coincidir con los parámetros que se deben setear en el nodo Gateway. Estos parámetros son:

PARAMETRO	VALOR	FUNCIÓN
Mode	1	Determina el ancho de banda a usar, el factor de dispersión entre otras cosas.
APP KEY	Clave compuesta por números	Permite que un nodo Gateway reciba únicamente mensajes de un nodo sensor cuya clave de aplicación (APP KEY) está registrada en el archivo keys.py en el nodo Gateway.
Freq	433.3	Indica la frecuencia en la que se va a transmitir la información.
Node_addr	Numero entero positivo	Se debe indicar un número que representa el identificador del nodo señor, por lo que debe ser único.

Tabla 16 - Parámetros del nodo Sensor

A continuación se ilustra un cuadro donde se puede observar la relación entre el ancho de banda seleccionado y el factor de dispersión. Según sea la combinación de estos, se determinara cuanto tiempo estará la señal en el aire propagándose en base a la cantidad de bits a enviar. En resumen a menor ancho de banda y mayor factor de dispersión la señal permanecerá mayor tiempo en el aire pero permitirá un mayor alcance. Análogamente incrementando el ancho de banda y disminuyendo el facto de dispersión, la señal viajara más rápido pero con alcance más reducido.



				time on air in second for payload size of						
LoRa mode	BW	CR	SF	5 bytes	55 bytes	105 bytes	155 Bytes	205 Bytes	255 Bytes	max thr. for 255B in bps
1	125	4/5	12	0.95846	2.59686	4.23526	5.87366	7.51206	9.15046	223
2	250	4/5	12	0.47923	1.21651	1.87187	2.52723	3.26451	3.91987	520
3	125	4/5	10	0.28058	0.69018	1.09978	1.50938	1.91898	2.32858	876
4	500	4/5	12	0.23962	0.60826	0.93594	1.26362	1.63226	1.95994	1041
5	250	4/5	10	0.14029	0.34509	0.54989	0.75469	0.95949	1.16429	1752
6	500	4/5	11	0.11981	0.30413	0.50893	0.69325	0.87757	1.06189	1921
7	250	4/5	9	0.07014	0.18278	0.29542	0.40806	0.5207	0.63334	3221
8	500	4/5	9	0.03507	0.09139	0.14771	0.20403	0.26035	0.31667	6442
9	500	4/5	8	0.01754	0.05082	0.08154	0.11482	0.14554	0.17882	11408
10	500	4/5	7	0.00877	0.02797	0.04589	0.06381	0.08301	0.10093	20212

Tabla 17 - LoRa Mode: Variación de combinación de ancho de banda y dispersión de espectro

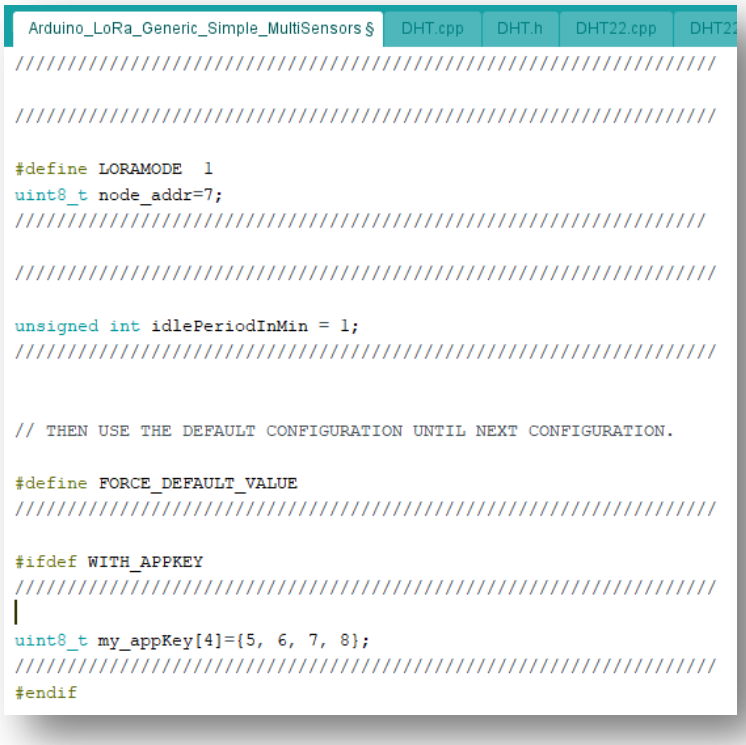


Figura 33 - Configuración de parámetros arduino

Una vez que el nodo sensor ya realizo las mediciones necesarias, envía la información al nodo Gateway para que este las suba al servidor. A continuación se ilustra la información que se envía al nodo Gateway separando los valores medidos mediante una barra "/". Una vez que se envía el paquete, arduino se pone en modo ahorro de energía (sleep mode).

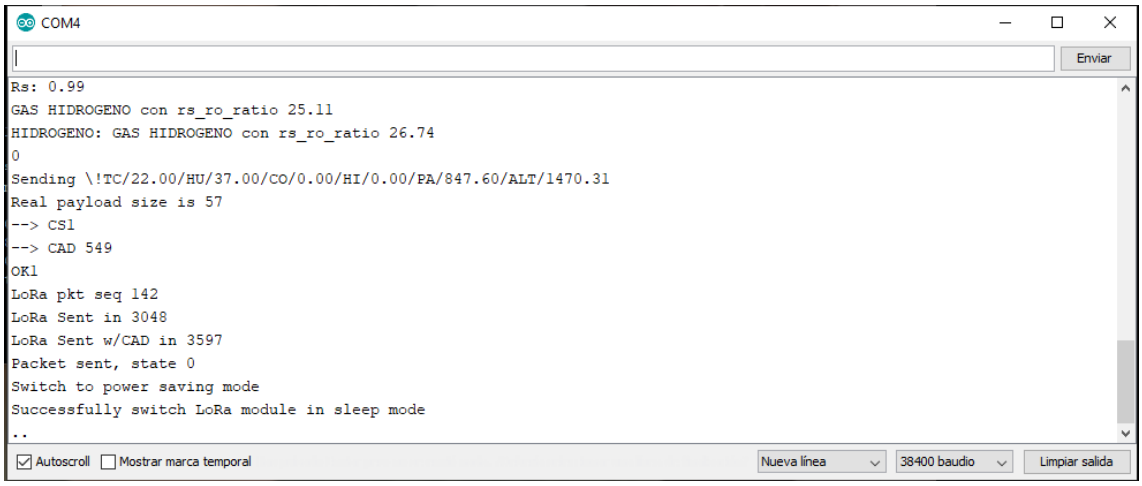


Figura 34 - Salida por puerto serie de la ejecución del software en arduino

La estructura del paquete está determinada de la siguiente manera:

dst addr (1 byte)	packet type (1 byte)	src addr (1 byte)	sequence number (1 byte)	payload data (variable length)

Tabla 18 - Estructura del paquete

- **dst addr:** Bit que determina la dirección de destino.
- **packet type:** Bit que determina el tipo de paquete. Se emplea para indicar si se usa una clave aplicación y si se envía cifrada la información.
- **src addr:** Dirección fuente, es el indicador (ID) del nodo sensor que está enviando el paquete.
- **payload data:** Es la carga del paquete, contiene toda la información restante. Aquí es donde se indican los resultados obtenidos por los sensores. Ejemplo TC/22.00 indica que se registró una temperatura de 22°C.

Comunicación desde el nodo Gateway

En el nodo Gateway también es necesario indicar la frecuencia en la que va estar escuchando. Para esto al encenderse corre un comando que pone en marcha el software que se encarga de recibir la información y enviarla al servidor para ser almacenada. El comando es

" ./loragateway –-mode 1 –-freq 433.3"

Donde **./loragateway** es el script que se encarga de poner el modulo LoRa del nodo Gateway a la escucha. Una vez que este recibe un paquete, lo que hace es buscar en la lista de keys.py , si el nodo que envió dicha información está registrado, de ser así, procede a ejecutar el script cloud.py el cual se encarga de parsear la información recibida y cárgala al servidor mediante el servicio REST empleando el protocolo HTTP.

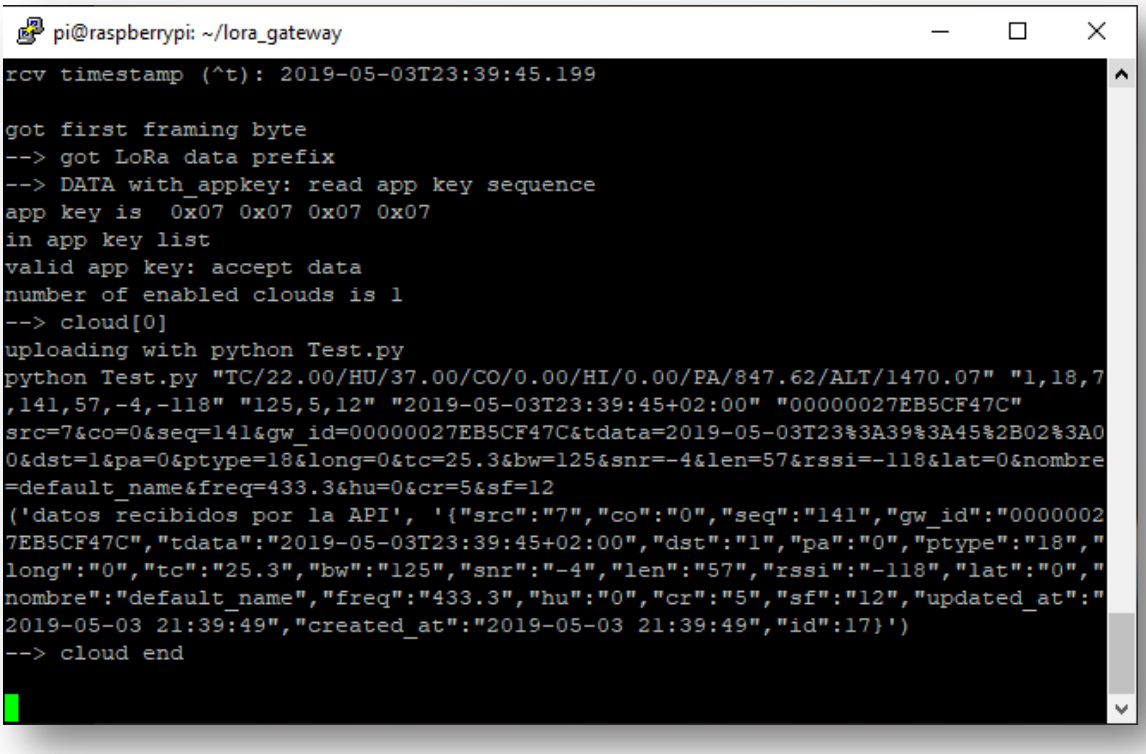


Figura 35 - Datos enviados al servidor desde el nodo Gateway

Cabe destacar que para que el nodo Gateway pueda enviar información al servidor mediante la API REST, este debe tener una autorización provista por el servicio REST mediante un TOKEN (cadena alfanumérica).

Herramientas Utilizadas

Apache

El Proyecto *Apache*[2] HTTP Server es un esfuerzo por desarrollar y mantener un servidor HTTP de código abierto para sistemas operativos modernos, incluidos UNIX y Windows. El objetivo de este proyecto es proporcionar un servidor seguro, eficiente y extensible que proporcione servicios HTTP sincronizados con los estándares HTTP actuales.

MySQL

Es un sistema de gestión de bases de datos relacionales cliente / servidor SQL originario de Escandinavia. MySQL incluye un servidor SQL, programas cliente para acceder al servidor, herramientas administrativas *Paul DuBois*[3].

XAMPP

Es una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl. El paquete de instalación de *XAMPP*[4] ha sido diseñado para ser increíblemente fácil de instalar y usar.

Laravel Lumen

Es micro-framework [5]para PHP creado por Taylor Otwell que comparte muchos de los componentes de su «hermano mayor» Laravel, pero Lumen es una versión más liviana de Laravel y orientado más a la creación de APIs y micro servicios "*Duilio Palacios, 2015*".

PHP

PHP [6](acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML "*The PHP Group, 2019"*.

---

[2] https://httpd.apache.org/
[3] MySQL: Developers libarary 4° Edition
[4] https://www.apachefriends.org/es/index.html
[5] https://styde.net/lumen-la-version-micro-framework-de-laravel-php/

PHPStorm

PhpStorm[7] es perfecto para trabajar con Symfony, Laravel, Drupal, WordPress, Zend Framework, Magento, Joomla, CakePHP, Yii y otros frameworks.

JQuery

Es una biblioteca de JavaScript rápida, pequeña y rica en funciones. Hace que cosas como el desplazamiento y la manipulación de documentos HTML, el manejo de eventos, la animación y Ajax sean mucho más simples con una API fácil de usar que funciona en una multitud de navegadores. Con una combinación de versatilidad y extensibilidad, *jQuery*[8] ha cambiado la forma en que millones de personas escriben JavaScript.

Bootstrap

Bootstrap[9], s un framework de código abierto para desarrollar con HTML, CSS y JS. Permite crear prototipos rápidamente o crear toda una aplicación con variables, sistema de cuadrícula receptiva, componentes pre compilados extensos y complementos potentes creados en jQuery.

Postman

Es una *herramienta*[10] que facilita y agiliza el desarrollo y prueba de APIs RESTs.

Putty

PuTTY[11] es un cliente SSH y telnet, desarrollado originalmente por Simon Tatham para la plataforma Windows. PuTTY es un software de código abierto que está disponible con código fuente y está desarrollado y respaldado por un grupo de voluntarios.

Raspbian OS

---

[6] https://www.php.net/manual/es/intro-whatis.php
[7] https://www.jetbrains.com/phpstorm/
[8] https://jquery.com/
[9] https://getbootstrap.com/
[10] https://www.getpostman.com/
[11] https://www.putty.org/

Raspbian[12] es el sistema operativo oficial compatible de la Fundación. Viene con una gran cantidad de software pre instalado para educación, programación y uso general. Tiene Python, Scratch, Sonic Pi, Java y más.

Git

Git[13] es un sistema de control de versiones, que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que se pueda recuperar versiones específicas más adelante.

Waziup

Es un proyecto de la Unión Europea-África que desarrolla tecnologías de IoT en África. "Wazi" es una palabra swahili para "abrir": *WAZI-UP* [14]abre nuevas tecnologías de IoT para los países africanos. Desarrolla IoT de vanguardia, comunicaciones rentables y tecnologías de big data para las necesidades de los países en desarrollo.

Lora Framework

Es un framework de uso libre, desarrollado por Conduc Pham[15]. Permite desarrollar proyectos que requieran integración de LoRa, de manera más sencilla y rápida.

Fritzing

Fritzing[16] es una iniciativa de hardware de código abierto que hace que la electrónica sea accesible como material creativo para cualquier persona. Ofrecemos una herramienta de software, un sitio web comunitario y servicios en el espíritu de Processing y Arduino, fomentando un ecosistema creativo que permite a los usuarios documentar sus prototipos, compartirlos con otros, enseñar electrónica en un aula y diseñar y fabricar pcbs profesionales.

Metodología

El desarrollo de la solución se hará siguiendo una metodología de desarrollo iterativa e incremental, a fin de permitir un desarrollo veloz, brindando al final de cada iteración una porción funcional de software. A medida que se van desarrollando las iteraciones, se van incrementando las funcionalidades del software y se va refinando.

---

[12] https://www.raspberrypi.org/downloads/raspbian/
[13] https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Acerca-del-Control-de-Versiones
[14] https://www.waziup.eu/about/
[15] https://github.com/CongducPham
[16] https://fritzing.org/home/

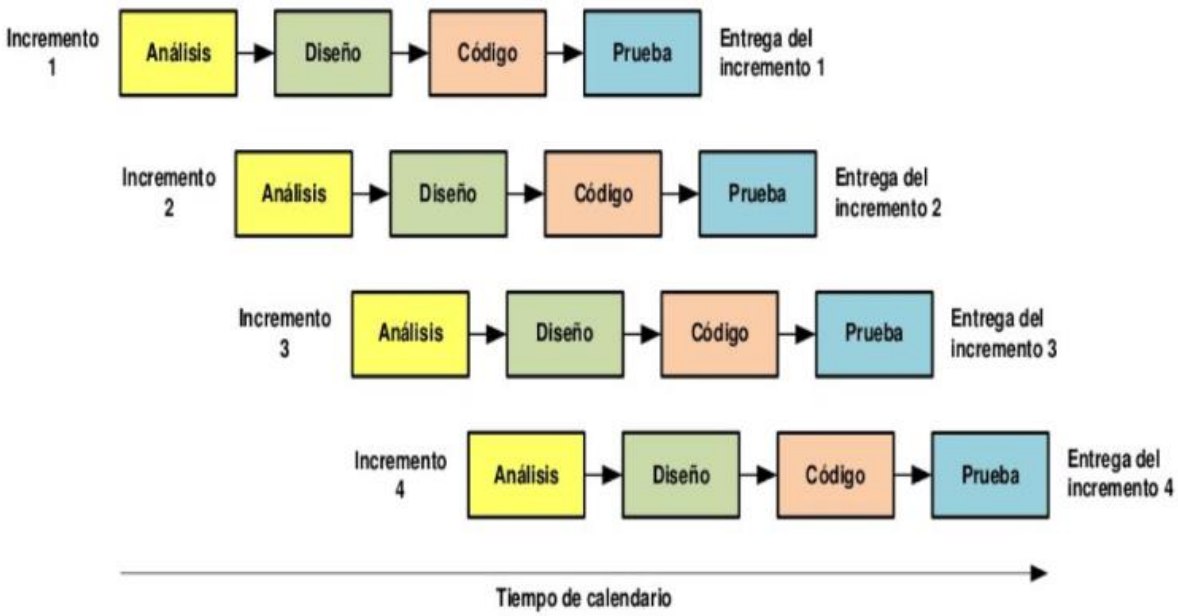


Figura 36 - Metodología Iterativa e Incremental

Para llevar a cabo el desarrollo del software se requieren de 3 incrementos:

1. **Realizar mediciones y comunicarlas al nodo Gateway:** haciendo uso del protocolo LoRa y de los lenguajes ya mencionados anteriormente se realizan las mediciones de los parámetros del aire y el ambiente, y se los envía al nodo Gateway.

2. **Comunicación entre los nodos Gateway's y el servidor:** mediante el uso del api rest, es posible que el nodo Gateway envíe los datos al servidor y este los almacene.

3. **Generar reportes:** haciendo uso del aplicativo web es posible generar y visualizar reportes de los datos obtenidos.

Las fases de análisis se detallan en cada iteración, mientras que la de diseño ya fue establecida previamente y las fases de código y prueba se adjuntan al proyecto.

Se confecciono una lista de eventos del sistema y una lista de casos de uso, que son de utilidad para las herramientas de análisis estructurado moderno usadas durante la fase de análisis.

Las herramientas que se emplearon en la fase de análisis son:

- **Lista de casos de usos**
- **Especificación de Casos de Uso:** Herramienta de UML que ayuda a la comprensión de los procesos y funciones.

Mientras que para la fase de diseño se hizo uso de:

- **Diagrama de actividades con responsabilidades**
- **Diagrama de CU**
- **Diagrama de clases conceptuales**
- **Diseño de interfaz**
- **Diseño de datos**

La lista de casos de usos se encuentra brevemente definida por:

Lista de Casos de Uso

1. Obtener y comunicar datos
2. Generar reportes

ITERACIÓN 1:

DISEÑO:

Diagrama de actividades con responsabilidades:

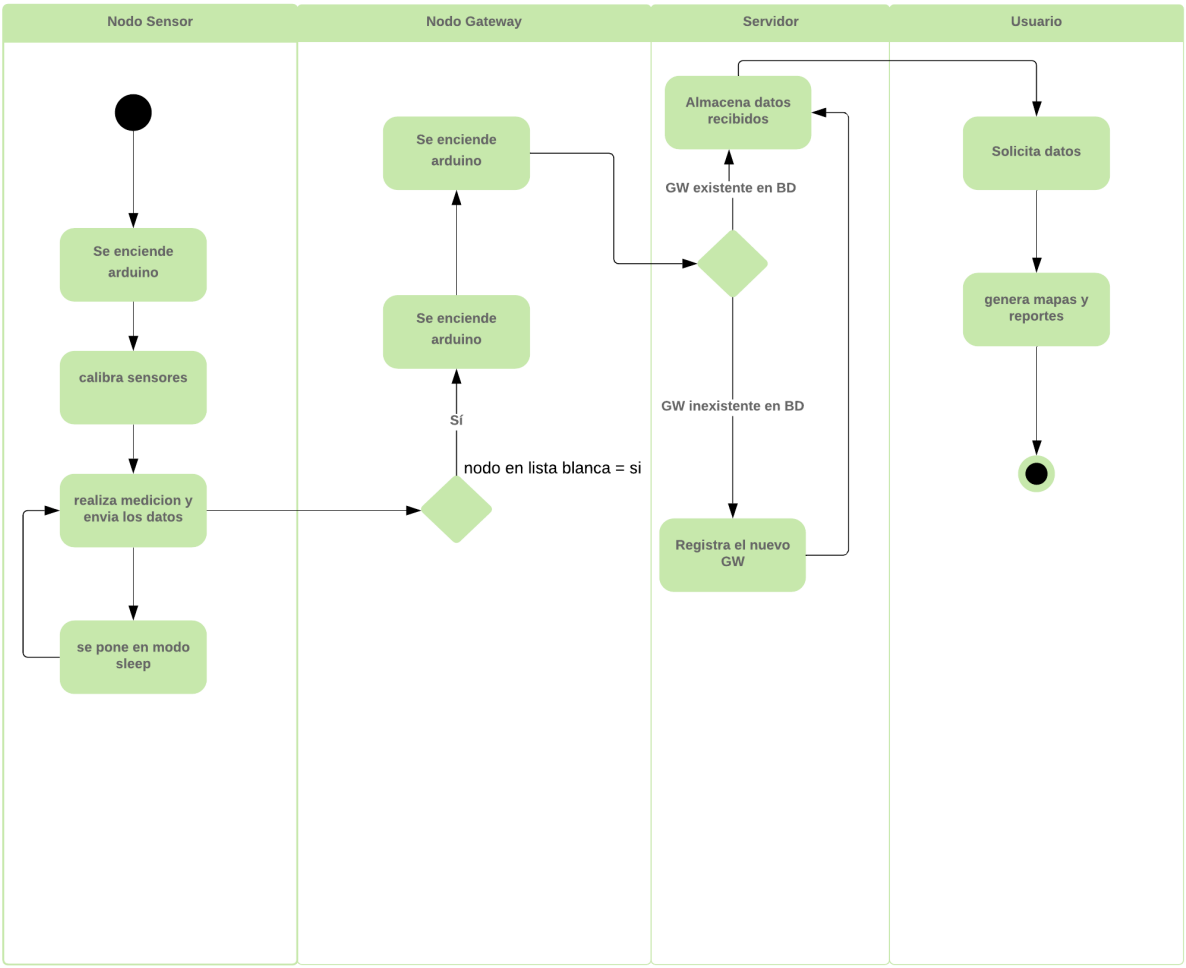


Figura 37 - Diagrama de actividades

Diagramas de casos de usos:

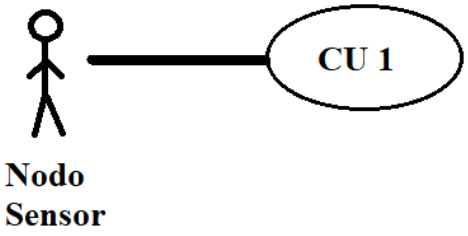


Figura 38 – Diagrama caso de uso 1

Diagrama de clases Conceptuales:

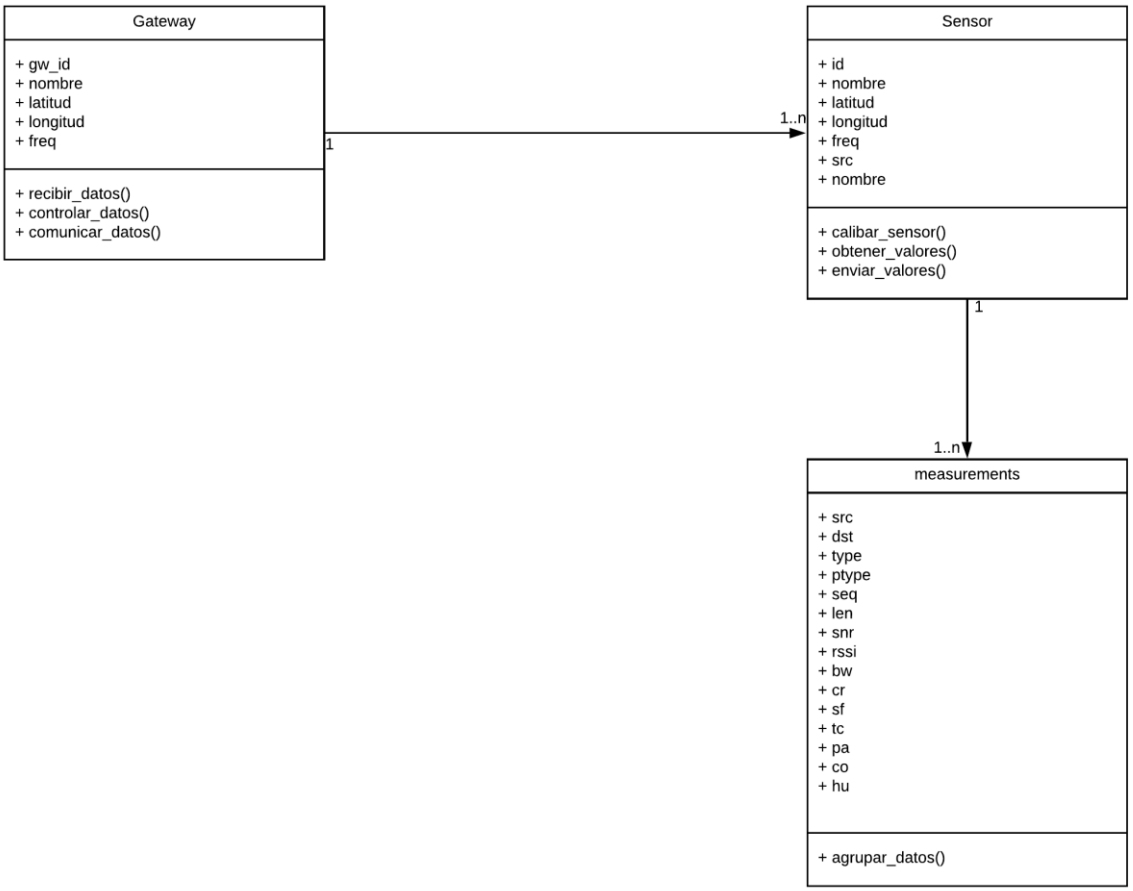


Figura 39 - Diagrama de clases conceptuales

ANALISIS:

Caso de uso: Obtener y comunicar datos
Actor: Nodo Sensor
Pre condiciones: Sensores activos
Post condiciones: Paquete de datos

Escenario principal de éxito

Actor	Sistema
1- Iniciar Nuevo C.U Obtener y comunicar datos 2- Recolecta información de sensores.	3- Procesa información. 4- Envia información al nodo gateway. 5- Nodo GW controla si esta en lista blanca el nodo sensor. 6- Comunica datos al servidor. 7- El servidor almacena los datos. 8- Fin C.U.

Escenarios alternativos

4- Envia información al nodo gateway falla

Actor	Sistema
	4.1 - Reintenta envio 10 veces. 4.2 – Mensaje de error en Puerto serie. 4.3 – Se coloca en modo ahorro de energia y espera el tiempo estiamdo para volver a repetir el proceso.

ITERACIÓN 2:

DISEÑO:

Diseño de datos:

En cuanto a la base de datos, el core es sencillo, empleando tres tablas principales y dejando de lado las tablas de usuarios y tablas propias del framework, el diagrama de entidad relación está definido de la siguiente manera:

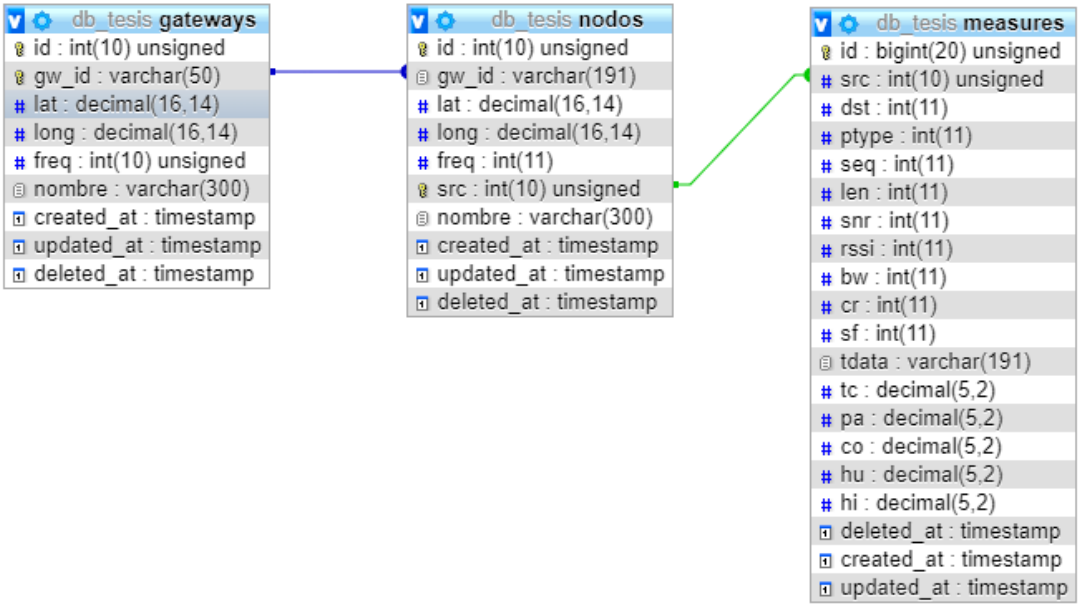


Figura 40 - DER

Un nodo GATEWAY puede poseer múltiples nodos sensores, a su vez, cada nodo sensor (simplificado el nombre de la tabla como sensor) registra múltiples mediciones (en ingles measurements).

Diseño de interfaz

El diseño de interfaz de las distintas pantallas que componen el sistema ya se mostró previamente *(ver páginas de la 56 a 63).*

Diagrama de casos de usos:

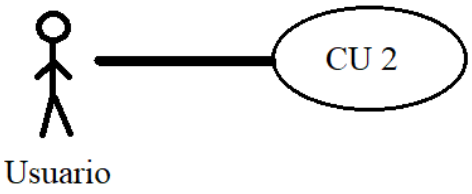


Figura 41 - Diagrama caso de uso 2

ANALISIS:

Caso de uso: Generar reporte
Actor: Usuario
Pre condiciones: Datos cargados previamente
Post condiciones: Reporte con graficas generado

Escenario principal de éxito

Actor	Sistema
1- Iniciar Nuevo C.U Generar reporte de nodo	
	2- Mostrar lista de nodos
3- Elige nodo	
	4- Obtiene y organiza datos
	1- Muestra informacion en vista
6- Visualiza reprote	
	7- Fin C.U

5- Resultados o verificación experimental

Experimento y prueba realizada

Hardware

Para comprobar el funcionamiento del sistema se realizaron diversas pruebas desde su construcción hasta su puesta en funcionamiento.

Inicialmente lo que se probó fue el hardware, para verificar que todo funcione correctamente. Para esto se probó por separado cada uno de los componentes.

- ✓ Se verifico que la placa arduino funcionara correctamente
- ✓ Se probó el sensor de temperatura y humedad DHT11
- ✓ Se realizaron pruebas y calibración del sensor de CO MQ-9
- ✓ Se probó el sensor de presión atmosférica bmp280
- ✓ Se realizaron pruebas de transmisión y recepción de los módulos LoRa SX1278
- ✓ Se probó el circuito construido en PCB y se determinó que el voltaje que se entrega a cada componente de este, es el correcto.

Lo primordial fue realizar la prueba de transmisión para determinar el alcance de los módulos. Para esto se empleó una batería de 9V para alimentar el circuito y se alejó el nodo sensor del nodo Gateway más de 400 metros.

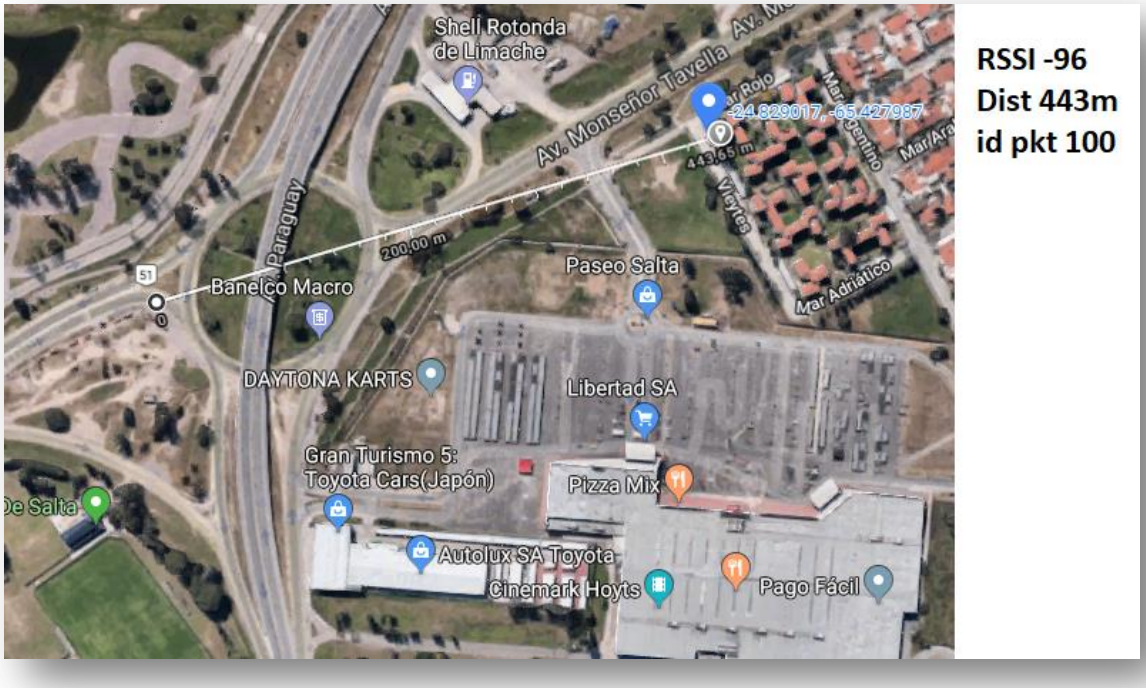


Figura 42 - Resultado de prueba de alcance modulo LoRa

Como se puede observar en la imagen, la distancia alcanzada en este experimento fue de 443m, donde el paquete fue recibido con un RSSI de -96. RSSI es el indicador de fuerza de la señal recibida, cuanto más cercano a 0 es el valor, mejor es. Según lo informado por el fabricante del módulo de transmisión LoRa, este tiene una sensibilidad de hasta -148dBm, por lo que se podría conseguir mayor alcance bajo otras condiciones. De hecho algunas personas realizaron pruebas en lugares más abiertos consiguiendo distancias superiores a los 4km.

También se realizó una prueba de la dirección en la cual soldar la antena al módulo LoRa para verificar si variaba o no la posición de esta y se obtuvo que varía pero en proporciones muy pequeñas.

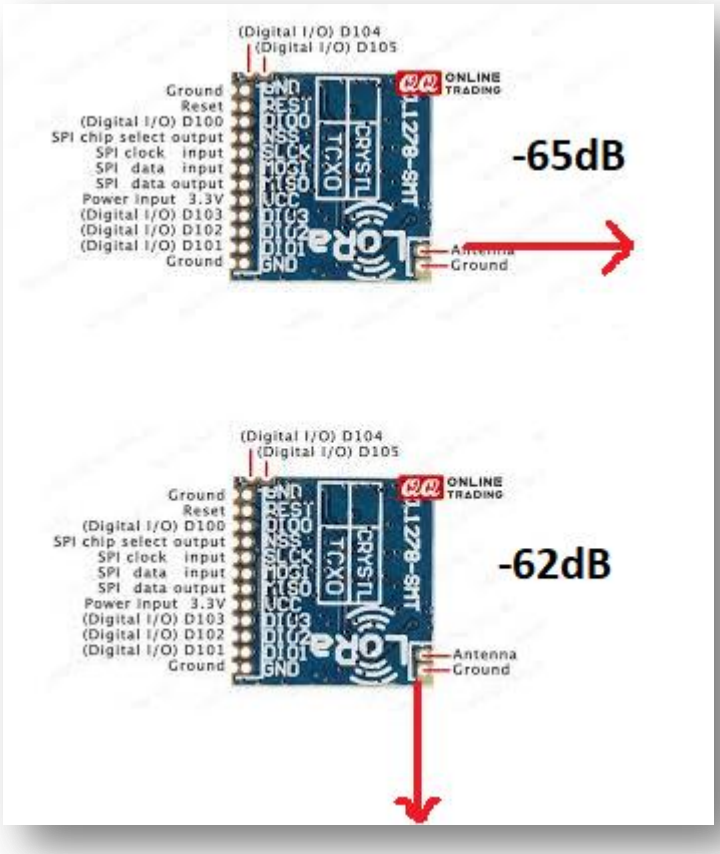


Figura 43 - Prueba de dirección de antena en modulo LoRa

Software

En cuanto al software, se controló que todo funcione correcto tanto en el nodo sensor como en el nodo Gateway y el servidor.

Para controlar que en el nodo sensor funcionara correctamente, al realizar las pruebas se lo conócete a una PC con el fin de poder visualizar en consola la salida de información por el puerto serie COM4. Con la consola fue posible hacer un seguimiento para determinar si los sensores arrojaban valores correctos, si se enviaba de forma correcta la información o si había algún indicador de error en esta.

Del lado del nodo Gateway, también se empleó la consola del sistema operativo para observar el funcionamiento del mismo. Esto permitía saber si el sistema arrancaba de forma adecuada, si el modulo estaba configurado y funcionando correctamente y si se recibía algún paquete de información. También hacia posible saber mediante consola la respuesta del servidor, para saber si se produjo algún error al intentar cargar los datos.

Para probar el API REST desarrollado, se hizo uso de la herramienta POSTMAN, con el fin de agilizar el testing y depurar los distintos errores que se podría producir.

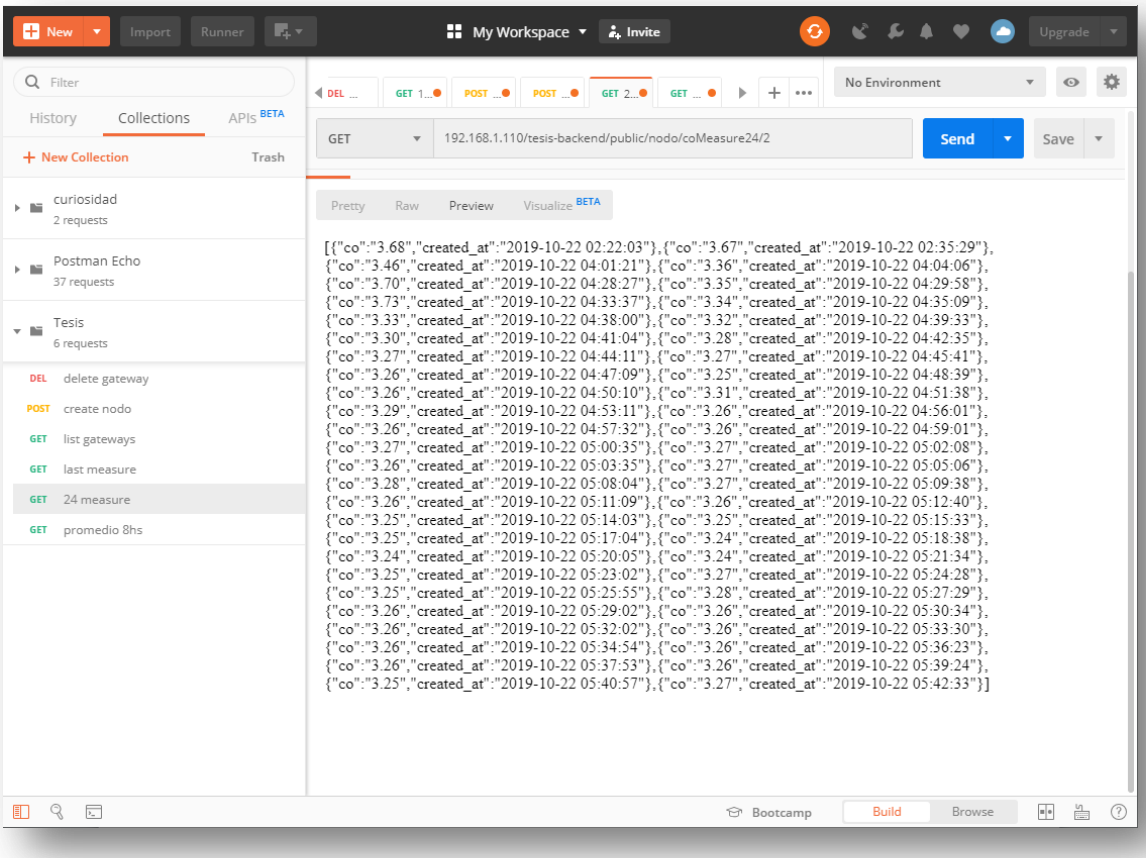


Figura 44 - Pruebas de API REST con POSTMAN.

Por ultimo para realizar la prueba del aplicativo web, se empleó el depurador de Google Chrome, el cual cuenta con diversas herramientas como consola de JavaScript, analizador de elementos y archivos, seguimiento de solicitudes HTTP.

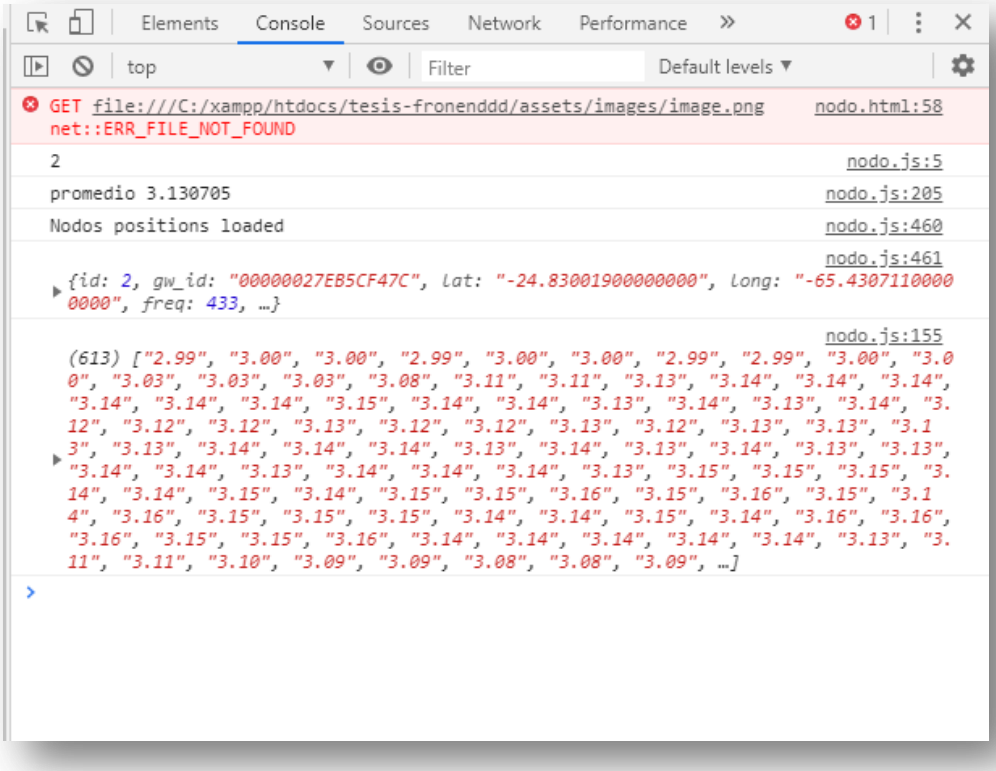


Figura 45 - Herramienta de depuración de Google Chrome

Cálculo de consumo del circuito

Se realizaron algunos cálculos para determinar la autonomía del nodo señor alimentado con baterías o pilas.

Primeramente se debe analizar cuanto consume cada componente del circuito, la sumatoria de estos consumos da como resultado el consumo total por hora del circuito completo.

El cálculo del consumo en miliamperios por hora del circuito está definido por:

$$\frac{cantidad\ de\ eventos\ por\ hora\ X\ Duración\ de\ cada\ evento}{3600\ segundos}\ X\ Consumo\ total\ de\ dispositivos$$

DISPOSITIVO	CONSUMO EN mA
Arduino	20
DHT11	2,5
MQ-9	150
Módulo LoRa SX1278	12
BMP280	0,003
	TOTAL: 184,503

Tabla 19 - Consumo de componentes del circuito

Si tenemos en cuenta que realizamos mediciones cada 2 minutos, con una duración promedio de 8 segundos, el consumo en mAh del circuito es de 12,30.

Fuente de Alimentación	Capacidad en mA
Batería 9V	500
4 Pilas AA 1,5	1700-3000
Batería Lipo	5300

Tabla 20 - Tipos de fuentes y capacidades

Una vez determinado el consumo por hora del circuito, se procede a calcular cuantas horas de autonomía brindan las distintas opciones de fuente, el cálculo se realiza de la siguiente manera:

$$Autonomía\ del\ Nodo\ Sensor = \frac{Capacidad\ de\ la\ fuente\ en\ mA}{Consumo\ en\ mAh\ del\ circuito}$$

- Para alimentar con una batería de 9V la autonomía del nodo sensor es de 40,65 horas.
- Para alimentar con 4 pilas alcalinas AA 1,5 la autonomía del nodo sensor es de 243,90 horas.
- Para alimentar con batería Lipo 5300mAh 11,1V la autonomía del nodo sensor es de 430,89 horas.

El ideal sería 4 mediciones por hora, dando un total de 96 mediciones por día, con un consumo de 1,64 mAh:

- Para alimentar con una batería de 9V la autonomía del nodo sensor es de 304,87 horas.
- Para alimentar con 4 pilas alcalinas AA 1,5 la autonomía del nodo sensor es de 1875 horas.
- Para alimentar con batería Lipo 5300mAh 11,1V la autonomía del nodo sensor es de 3231,7 horas

Resultados obtenidos

Tras dejar el sensor en funcionamiento durante al menos 2 semanas, se logró realizar más de 10.080 mediciones, con el fin de obtener información suficiente para comprobar si los datos obtenidos son aceptables y concuerdan con lo esperado.

Mediante el grafico que proporciona el sistema, se pudo determinar que los valores de CO que se registraron en la zona, tienen una forma de onda sinodal, registrando máximos y mínimos durante distintos periodos del día.

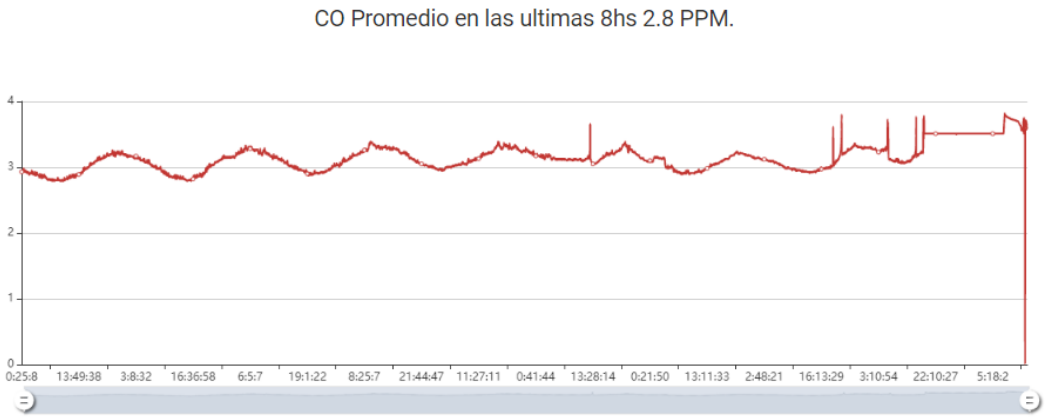


Figura 46 - Oscilación de CO en una semana

El grafico permite hacer un acercamiento para poder observar bien en que franjas horarias se registraron los niveles más altos y más bajos de CO. Obteniéndose niveles altos CO durante la mañana entre las 6:30 y las 8:30 horas, y niveles más bajos por las tardes entre las 17:00 y 19:30 horas.

En primera instancia se pensó lo más lógico, que se registren niveles más altos de CO en los horarios en que incrementa el tránsito a la zona, ya que muchas personas se desplazan en sus medios de transporte al trabajo, escuela, etc. Así es como se registró durante la mañana, hora en que incrementa el tránsito, pero que pasa con los valores mínimos, que se registraron durante las tardes, en el horario en el que el tráfico se reactiva nuevamente. Es en este punto donde se analizó la influencia de la temperatura, lo cual según establece la ley de Charles ***"siempre que la masa y la presión de un gas permanecen constantes, el volumen del gas es directamente proporcional a su temperatura absoluta."*** Lo que indica que el volumen de un gas es directamente proporcional a la temperatura absoluta de este.

$$V = k \, . \, T$$

Donde V es el volumen, T la temperatura absoluta y k la constante de proporcionalidad.

En resumen a mayor temperatura, mayor volumen, con lo cual las partículas de CO se encuentran más dispersas en el aire y es por esto que se deduce la disminución de nivel de CO durante la tarde, momento en el que la temperatura alcanza su pico máximo. A esta relación indirecta entre la temperatura y los niveles de CO que se registraron, se la pueda apreciar mediante los gráficos que proporciona el sistema.

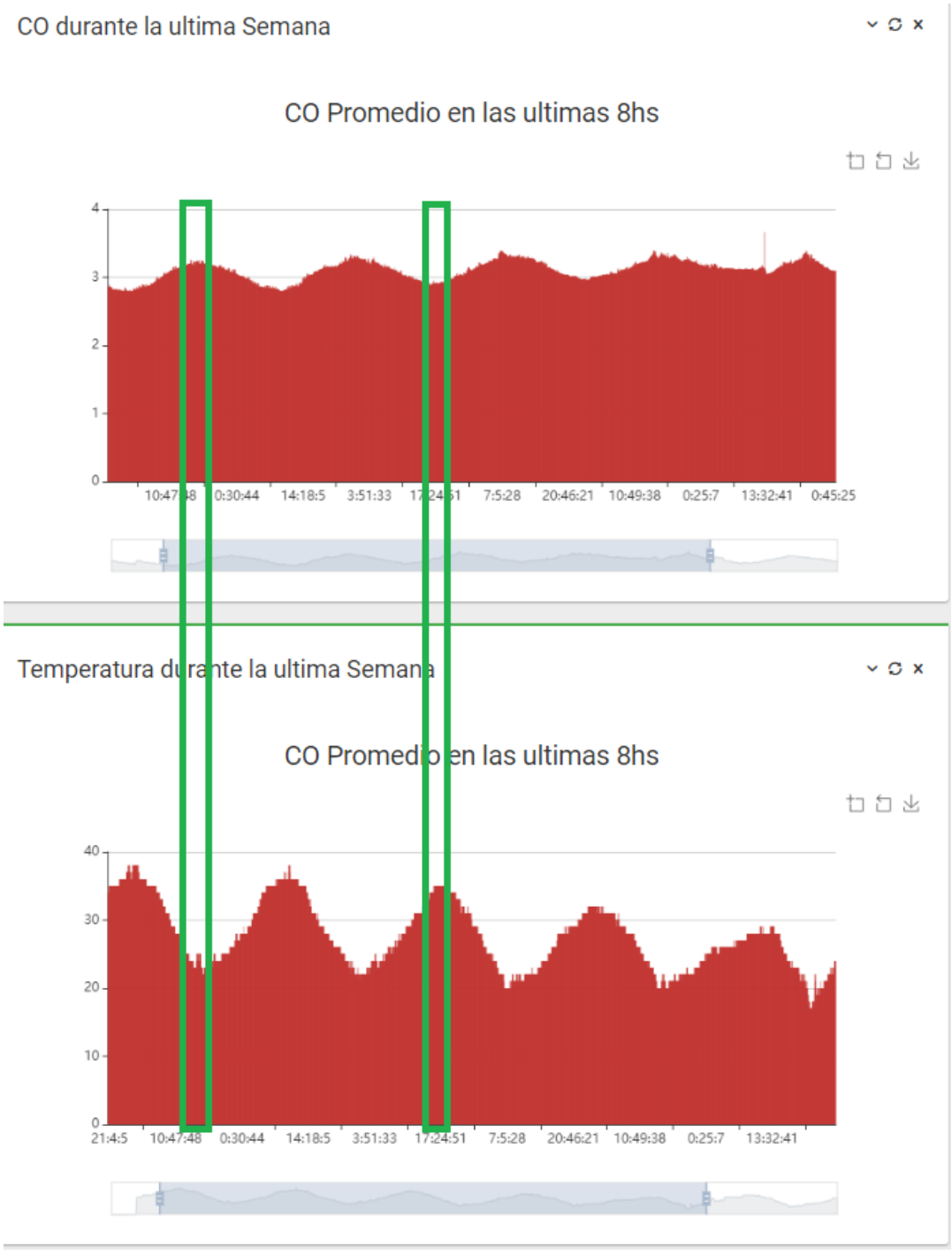


Figura 47 - Relación CO y temperatura

Figura 48 - Zoom en el gráfico de Temperatura y CO

Franja horaria en la que se registra altos niveles de CO y baja temperatura.

Esta relación se puede validar con datos registrados por otros proyectos como el de AQI[17] si bien la zona no es la misma, sirvió para constatar que se cumple dicha relación.

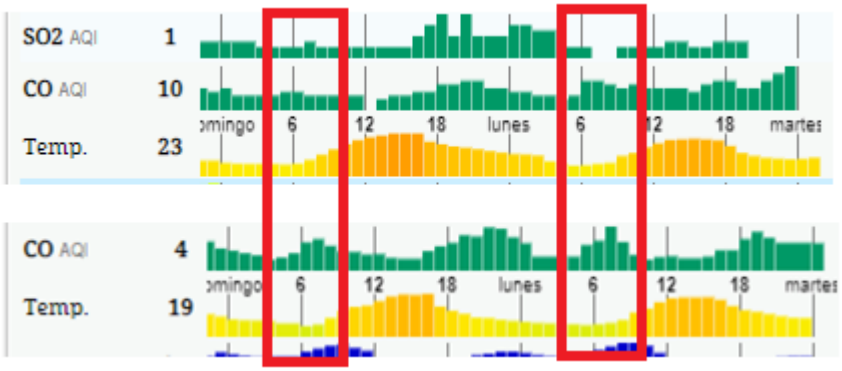


Figura 49 - Niveles de CO y Temperatura - Brasil

[17] http://aqicn.org Air Queality Index

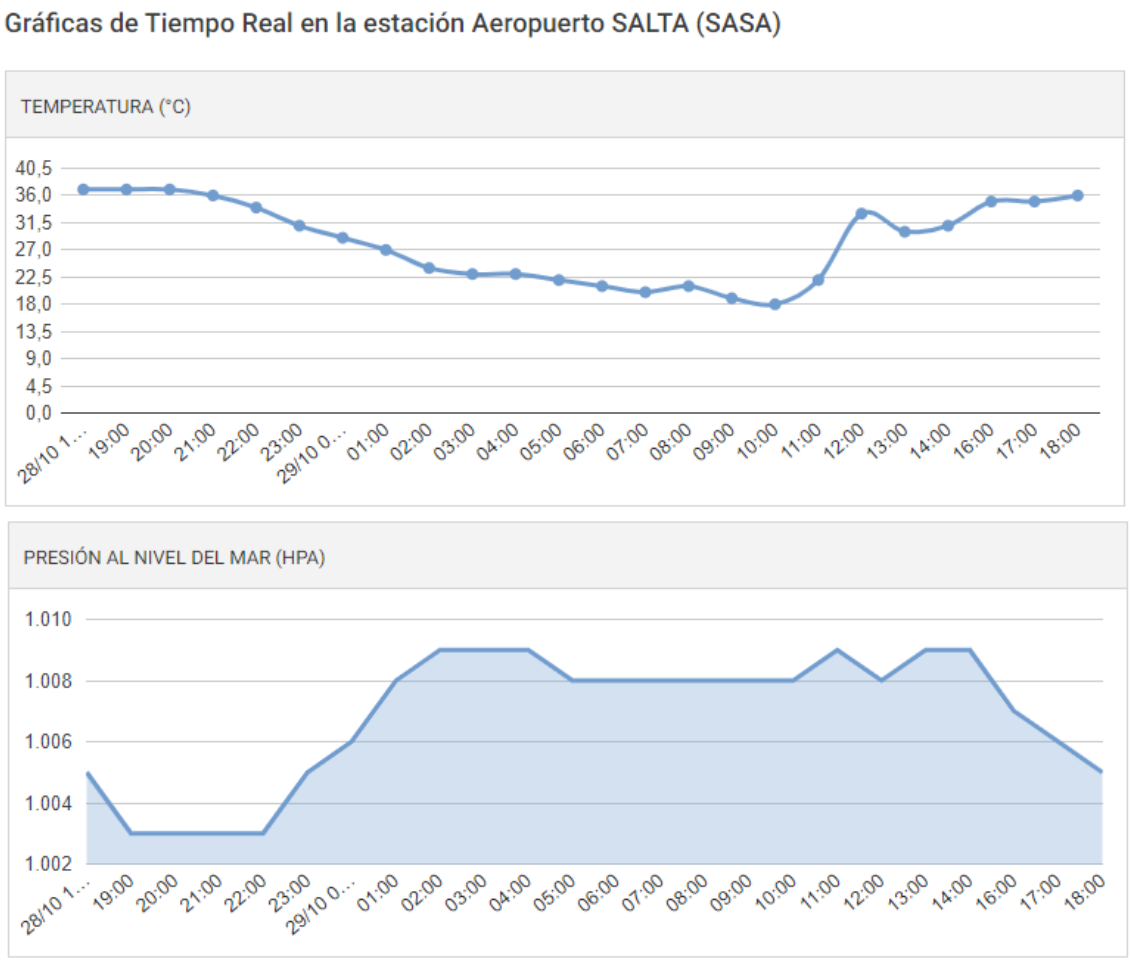


Figura 50 - Temperatura y presión atmosférica salta

La presión atmosférica también incide en la detección del CO, si bien se puede comparar de qué varía y no es constante, lo hace en pequeñas proporciones.

Por último se empleó la calculadora[18] que proporciona el AQI, que mediante la cantidad de CO en ppm calcula el índice para determinar la calidad del aire, para de esta manera validar los valores obtenidos.

[18] https://airnow.gov/index.cfm?action=airnow.calculator

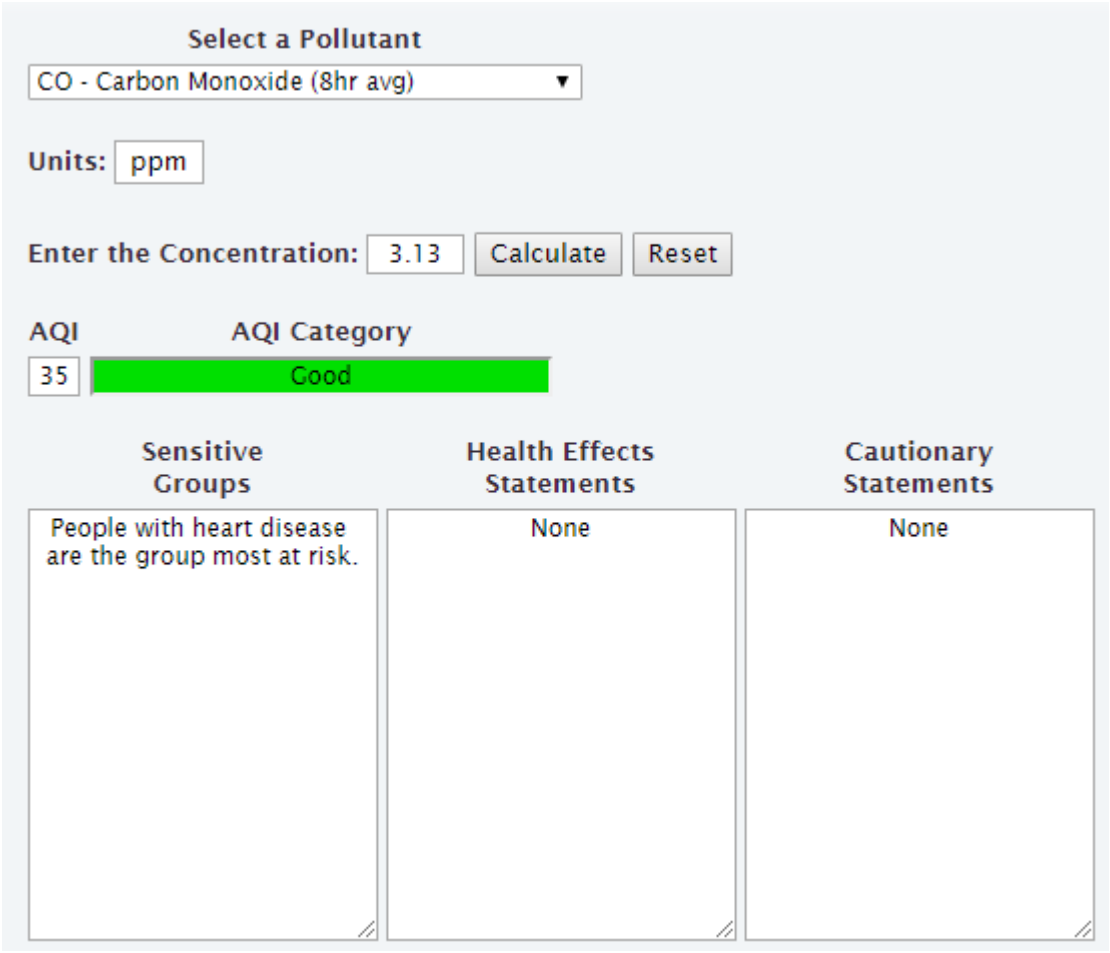


Figura 51 - Calculadora ICA

Para un promedio en 8 horas de 3.13 ppm se obtiene un índice de 35, el cual es menor de 50, por eso se determina que la calidad es buena, según se detalló en la tabla ICA anterior mente.

Dificultades no previstas encontradas y sus soluciones

Donde más problemas se encontraron, fue durante el diseño y armado del circuito, dado a la poca experiencia en electrónica entre otras cosas, como ser la habilidad para soldar componentes.

Como se puede apreciar en la figura 8, el modulo transceptor LoRa es muy pequeño, por lo que dificultó el soldado de cables a los pines, ya que estos son más pequeños que los estándar. Afortunadamente se contaba con 2 unidades más, por lo que estas fueron soldadas con mayor prolijidad.

Luego se presentaron problemas de alimentación, ya que algunos componentes no recibían el voltaje necesario. Esto se solucionó realizando mediciones del voltaje entregado con un multímetro y empleando conversores de voltajes de 5V a 3.3V ya que el arduino que se usó opera con 5V por lo que la salida del pin de alimentación para el circuito podría quemar aquellos sensores o el modulo lora que operan con 3.3V.

Por ejemplo, al intentar transmitir los datos del nodo sensor, se obtenía el mensaje lora *"send packet state 1"*, o cuando iniciaba el modulo indicaba *"set state 1"* mientras que el valor correcto es 0. La solución fue verificar y controlar que el módulo recibiera los 3.3V que requiere para su funcionamiento.

STATE	INTERPRETACIÓN
0	El comando se ejecutó sin errores
1	Se produjo un error mientras se ejecutaba el comando
2	El comando no fue ejecutado
3	El paquete se envió pero no se recibió un ACK

Tabla 21 - Posibles errores en el nodo Sensor

Otro problema que se presento fue que los sensores ocasionalmente fallaban, enviaban información errónea o nula. Lo primero que se hizo fue controlar el voltaje para comprobar que estén recibiendo la cantidad de energía correcta, al verificar esto y tras varios minutos buscando la falla, se determinó que el error estaba en los cables usados, los cuales a simple vista se encontraban en perfectas condiciones pero al parecer internamente tenían cortes.

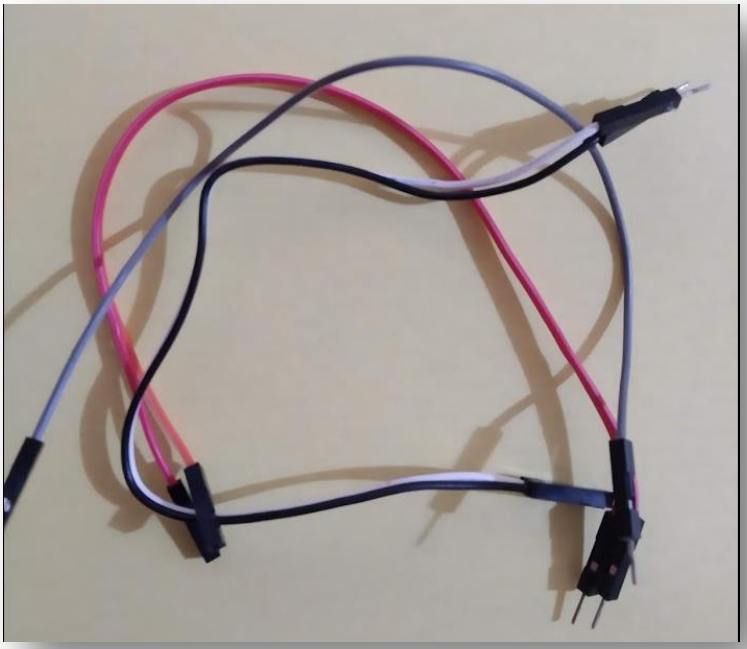


Figura 52 - Cables monofilamento para protoboard

La solución fue remplazarlos por cables multifilamento que son más flexibles.

Otro problema que se presento fue que el sensor de CO para que realice mediciones más precisas debía estar calentado al menos 48hs, para esto se lo conecto directamente a la salida de energía que alimenta el circuito, pasando antes por el nivelador de voltaje para reducir los 9V de la batería a los 5V que necesita el sensor. De aquí surge otro problema, y es la cantidad de energía que se disipa en forma de calor, energía que no es aprovechada y que agota la vida de la batería de forma más acelerada. Por eso lo ideal es o bien realizar menos mediciones diarias y encender el sensor de CO al momento de realizar la medición, o mantener el circuito conectado a una fuente de alimentación continúa de 9V mediante un transformador.

Por último, no se encontró información sobre cada cuanto tiempo se debe censar el aire, solo que se emplea el promedio en 8hs. Es por esto que se eligió realizar mediciones cada 2 minutos teniendo así 30 mediciones por hora, lo que da un total de 240 cada 8 horas y 720 por día. Lo ideal sería realizar cada 10 o 15 minutos cuando el nodo se alimenta con baterías o panel solar, para poder ahorrar más energía.

Lo ideal sería un panel solar que alimente una batería recargable para maximizar la autonomía del nodo sensor.

6- Conclusiones y futuras líneas de investigación

Este proyecto concluye con la construcción de un prototipo funcional, el cual puede ser implementado como una alternativa de bajo presupuesto a los sistemas de control de calidad del aire que se pueden encontrar en otras partes del país.

El desarrollo del mismo me permitió aplicar varios de los conocimientos adquiridos a lo largo de la carrera, como así también profundizar en otros y aprender cosas nuevas. Al contar con conocimientos muy básicos de electrónica, me encuentro satisfecho con el prototipo construido, pero sabiendo que aún se puede mejorar y optimizar.

Como futuras líneas de investigación queda:

- El desarrollo de una red con topología de malla que permita extender aún más la red, reduciendo el número de nodos Gateway, ya que cada uno de estos tiene la capacidad controlar cientos de nodos sensores y así también, permitir disminuir los costos.
- El desarrollo de un circuito optimizado para poder ser producido en masa, abaratando costos y facilitando el ensamble del mismo.
- Como el proyecto tiene como fin promover el cuidado del medio ambiente, es de suma importancia que los nodos sensores funcionen con energías renovables.
- Desarrollo de un aplicativo móvil para alertar cuando las emisiones de gases tóxicos superen un umbral indicado.
- Elaborar un mapa donde se visualice de forma sencilla la contaminación del aire.

7- Bibliografía

Ph. D. Roger S. Pressman, 2010. Ingeniería de Software. Un enfoque práctico Séptima edición. *Páginas 805. The McGraw-Hill Companies, Inc. Librería ISBN: 978-607-15-0314-5*

Índice de calidad del aire básico – *Recuperado el 25-08-2019 de https://www.airnow.gov/index.cfm?action=aqibasics.aqi*

PiGateway – *Recuperado el 12-08-2019 de http://cpham.perso.univ-pau.fr/LORA/RPIgateway.html*

Calibración Sensor MQ-9 – *Recuperado el 02-09-2019 de https://jayconsystems.com/blog/understanding-a-gas-sensor*

Calibración MQ-9 – *Recuperado el 02-09-2019 de https://learn.sparkfun.com/tutorials/hazardous-gas-monitor/all*

Guía Alimentación Arduino – *Recuperado el 08-09-2019 de https://aprendiendoarduino.wordpress.com/2016/11/09/alimentacion-arduino/*

Technical Data MQ-9 Gas Sensor – *Recuperado el 23-08-2019 de http://www.datasheet.es/PDF/904644/MQ-9-pdf.html*

Laravel Lumen Documentation – *Recuperado el 14-09-2019 de https://lumen.laravel.com/docs/6.x*

JQuery API – *Recuperado el 21-09-2019 de https://api.jquery.com/*

Tomasi Wayne, 2003. Sistemas de Comunicaciones Electrónicas 4ta Edición. *ISBN: 970-26-0316-1*

7- Anexos

Anexo 1 Instalación y configuración del nodo Gateway

Primero se debe descargar la ISO[19] con el sistema operativo provisto por Condugc Pham.

Una vez descargada, se debe montar en la tarjeta micro SD, para esto se empleó el software BalenaEtcher[20]. Una vez iniciado, se debe seleccionar la ISO, luego se debe seleccionar la tarjeta micro SD que se insertó y por último se debe presionar el botón flash y esperar a que se grabe correctamente.

[19] https://github.com/CongducPham/
[20] https://www.balena.io/etcher/

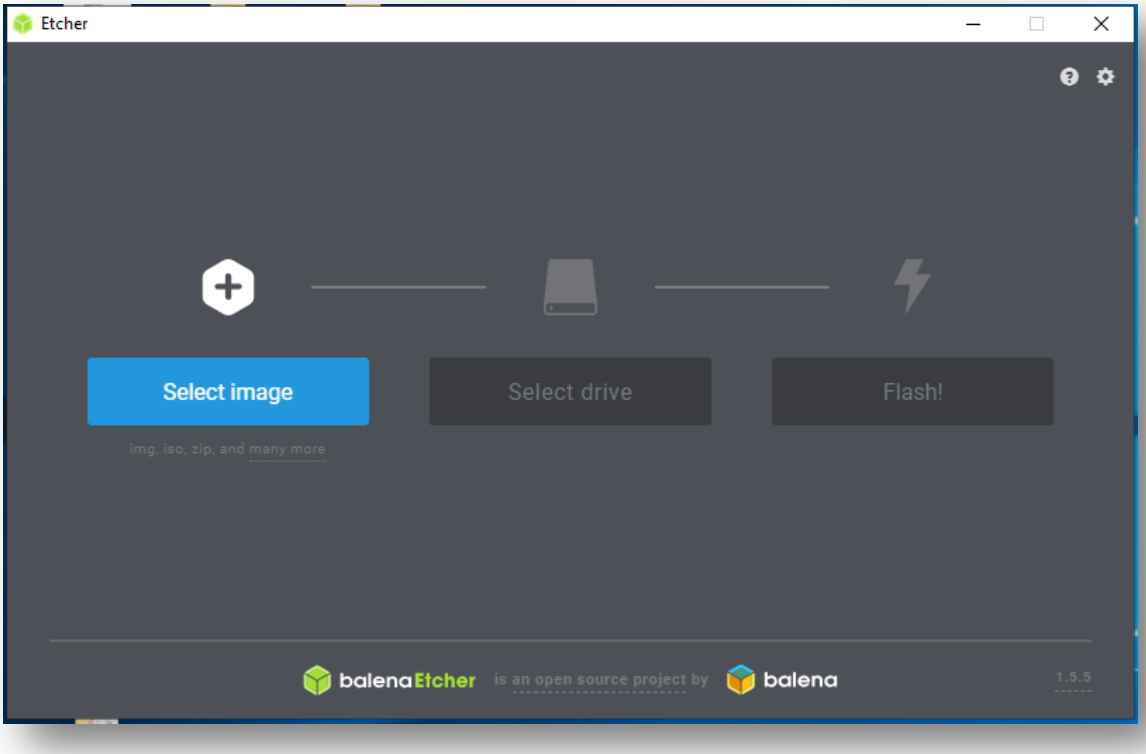


Figura 53 - Instalación ISO

Luego se debe colocar la tarjeta micro SD en la Raspberry Pi y encenderla. Una vez encendida, ya se puede acceder mediante WIFI, ya que inicialmente está configurado como un punto de acceso WIFI. El nombre del AP WIFI tiene el formato ***WAZIUP_PI_GW_XXXXXXXXXX***, se ingresa la contraseña que por defecto es ***loragateway.*** La dirección IP de la raspberry es 192.168.200.1

Si se desea conectar a internet la Raspberry Pi, se debe configurar esta como cliente WIFI y no como AP. Para esto se debe acceder a la configuración mediante el aplicativo web que trae instalado. Se procede abriendo el navegador y colocando la dirección IP de la Raspberry Pi 192.168.200.1.

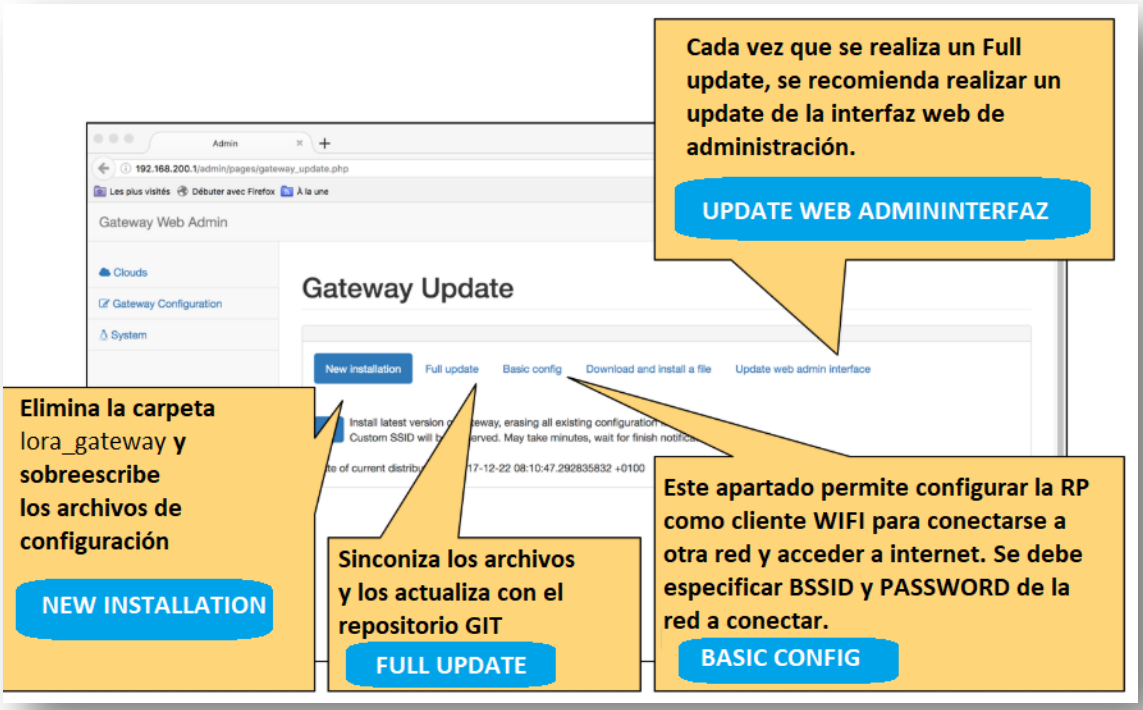


Figura 54 - Interfaz web configuración Raspberry Pi Gateway

Se debe seleccionar la opción **Basic config** , luego se debe cargar en el formulario el nobre de la red WIFI, es decir el BSSID y la contraseña de la misma. Una vez hecho esto se debe reiniciar la Raspberry Pi.

Al reiniciar, automáticamente se conectara a la red WIFI indicada *(Siempre y cuando se hayan indicado los datos correctamente, de lo contrario se deberá iniciar todo el proceso nuevamente ya que queda inaccesible el dispositivo).* Es necesario contar con un cliente SSH para poder acceder nuevamente a la Raspberry, esto se puede hacer desde otra PC o desde un smartphone. En el caso de hacerlo mediante un Smartphone, existen aplicaciones para esto,

una de ellas es JuiceSSH[21] y si se hace desde una PC se puede usar el que trae por defecto algunas distribuciones Linux, en Windows se puede instalar la herramienta PUTTY.

Una vez que se cuenta con alguna de estos clientes SSH, se procede a conectar informado la dirección IP de la Raspberry y el puerto que por defecto es el 22.

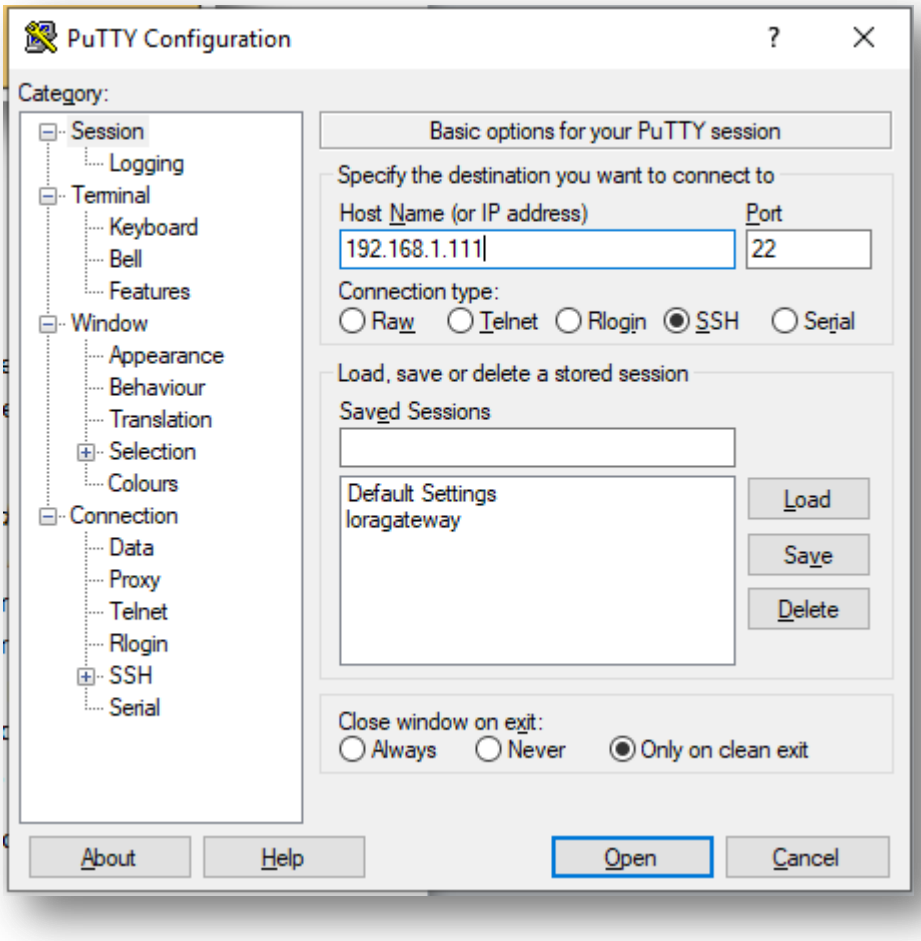


Figura 55 - Conexión SSH mediante PUTTY

[21] https://play.google.com/store/apps/details?id=com.sonelli.juicessh&hl=es_AR

Al conectarnos, se nos solicitara un usuario y una contraseña, que por defecto son

USUARIO	CONTRASEÑA
pi	loragateway

Es altamente recomendado cambiar ambos datos. Ya conectados, tendremos a disposición un menú con diferentes opciones para configurar y administrar el Gateway.

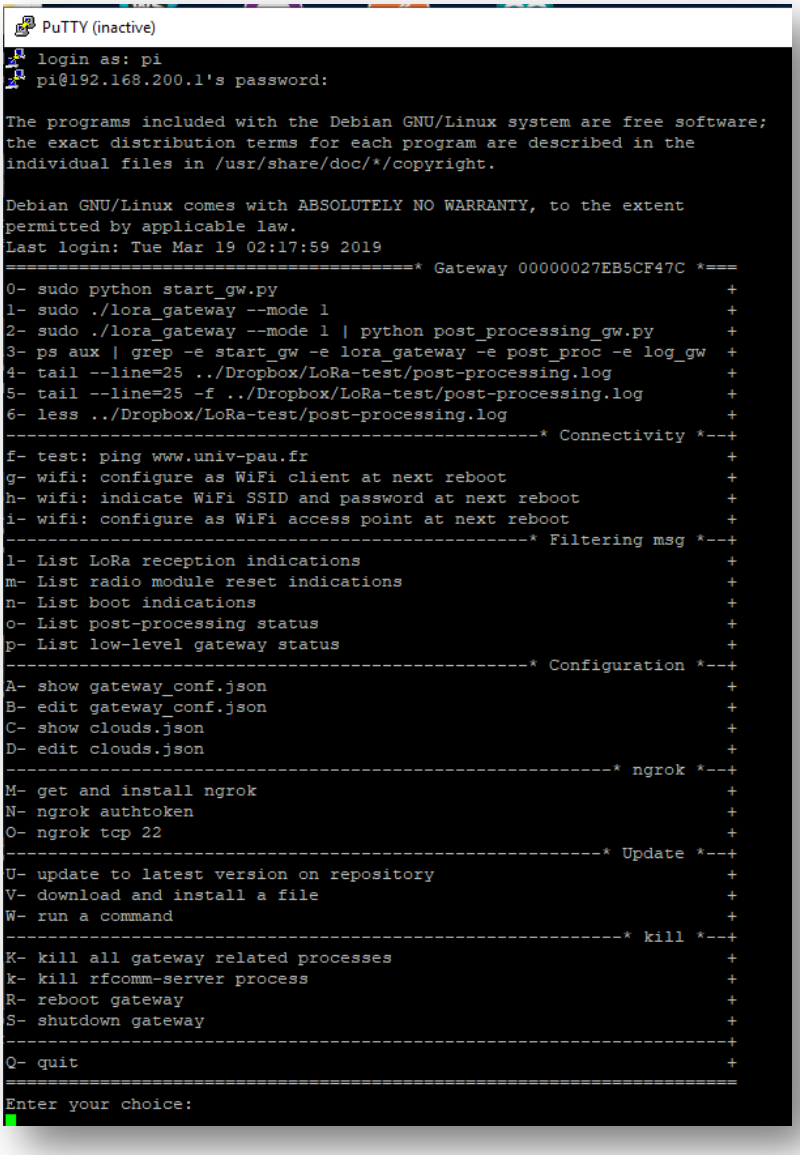


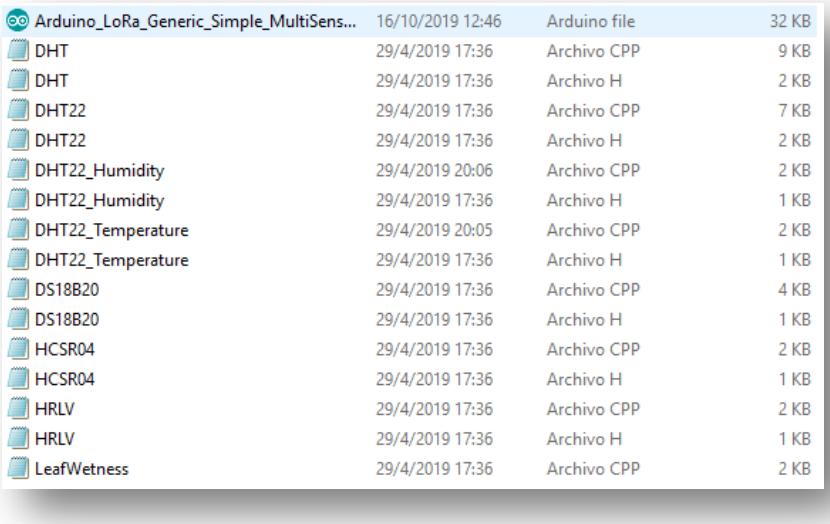
Figura 56 - Menú de configuración Gateway

Se debe seleccionar la opción B para configurar las opciones del Gateway, aquí es donde se configura el modo y la frecuencia, entre otras cosas. Cabe recordar que el modo a seleccionar es el número 1 y la frecuencia que se debe indicar es de 433.3.

Una vez realizado los cambios se selecciona la opción R para reiniciar el Gateway.

Anexo 2 Configuración Software de nodo Sensor

Primeramente se requiere descargar e instalar el IDE de Arduino[22]. Una vez hecho esto se requiere descargar el código fuente[23] que se cargara al arduino desde el repositorio GIT. Una vez descargado se abre la carpeta y se ejecuta el archivo *Arduino_LoRa_Generic_Simple_MultiSensors.*



Arduino_LoRa_Generic_Simple_MultiSens...	16/10/2019 12:46	Arduino file	32 KB
DHT	29/4/2019 17:36	Archivo CPP	9 KB
DHT	29/4/2019 17:36	Archivo H	2 KB
DHT22	29/4/2019 17:36	Archivo CPP	7 KB
DHT22	29/4/2019 17:36	Archivo H	2 KB
DHT22_Humidity	29/4/2019 20:06	Archivo CPP	2 KB
DHT22_Humidity	29/4/2019 17:36	Archivo H	1 KB
DHT22_Temperature	29/4/2019 20:05	Archivo CPP	2 KB
DHT22_Temperature	29/4/2019 17:36	Archivo H	1 KB
DS18B20	29/4/2019 17:36	Archivo CPP	4 KB
DS18B20	29/4/2019 17:36	Archivo H	1 KB
HCSR04	29/4/2019 17:36	Archivo CPP	2 KB
HCSR04	29/4/2019 17:36	Archivo H	1 KB
HRLV	29/4/2019 17:36	Archivo CPP	2 KB
HRLV	29/4/2019 17:36	Archivo H	1 KB
LeafWetness	29/4/2019 17:36	Archivo CPP	2 KB

Figura 57 - Código fuente arduino - Nodo Sensor

Una vez que se abre el IDE, se debe seleccionar en *Herramientas/Placa:* y se debe buscar Arduino Mini Pro o Pro Mini. De aquí en más restan realizar pequeñas modificaciones al código antes de cargarlo en la placa arduino. Primero se debe indicar la frecuencia en la que operara, para esto se quita el comentario eliminando las barras dobles "*//*" que anteceden *a #define BAND433*.

[22] https://www.arduino.cc/en/main/software
[23] https://github.com/matiast02/nodo_sensor.git

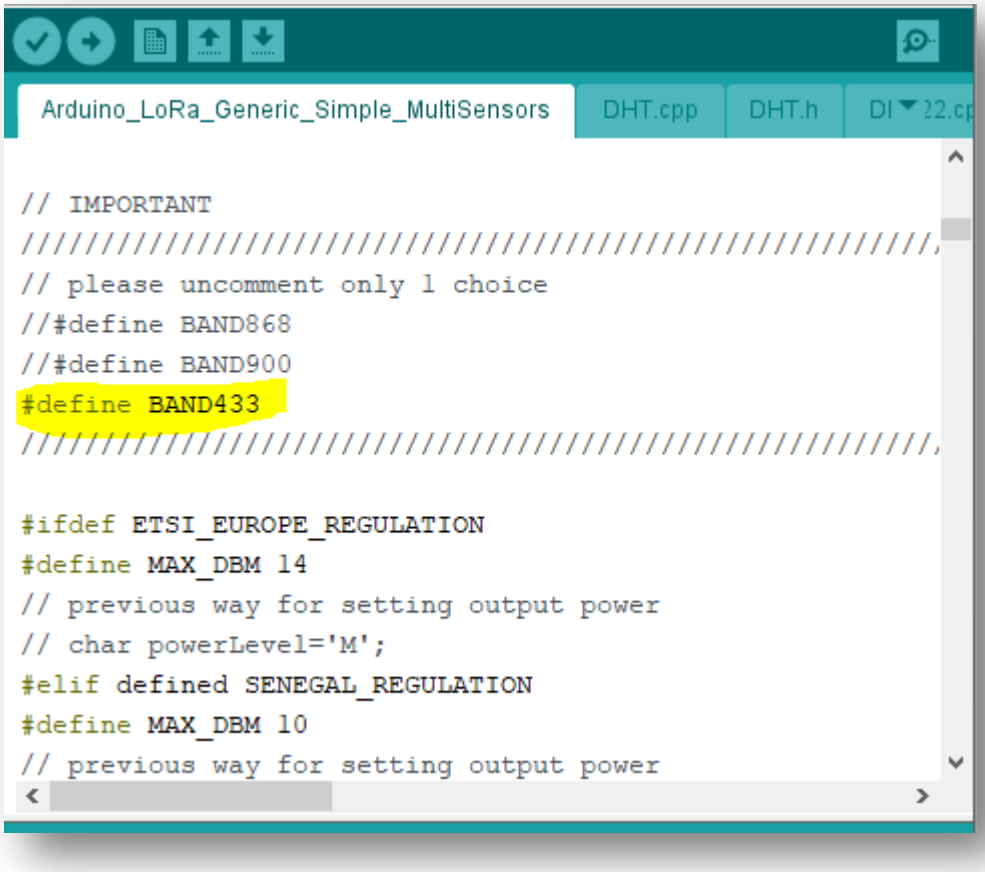


Figura 58 - Indicar frecuencia del nodo sensor

Luego se debe indicar el MODO LoRa en el que operara y la dirección que le corresponde al nodo. Es importante que el nodo tenga una dirección que no exista previamente en la red. La dirección esta definida por números enteros. También se debe indicar cada cuanto tiempo enviara los datos el nodo sensor al nodo Gateway.

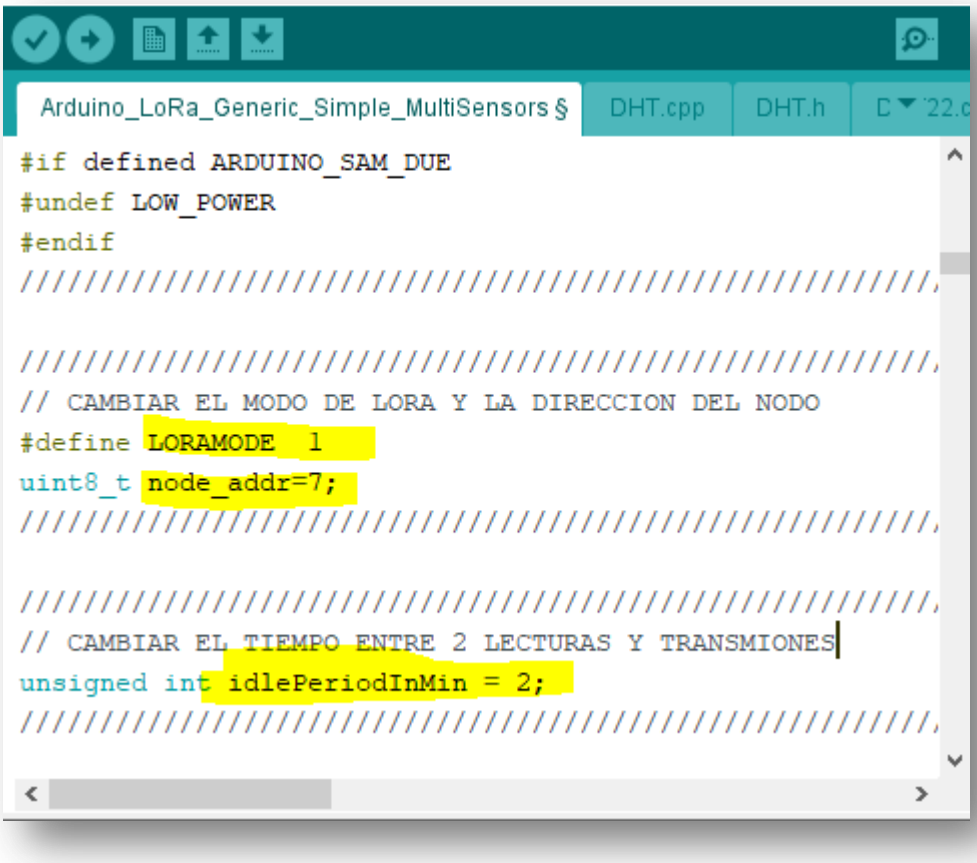


Figura 59 - Configuración de dirección, modo y tiempo

Por último se tiene que indicar la clave de aplicación, la cual debe estar en la lista Keys.py en el nodo Gateway.

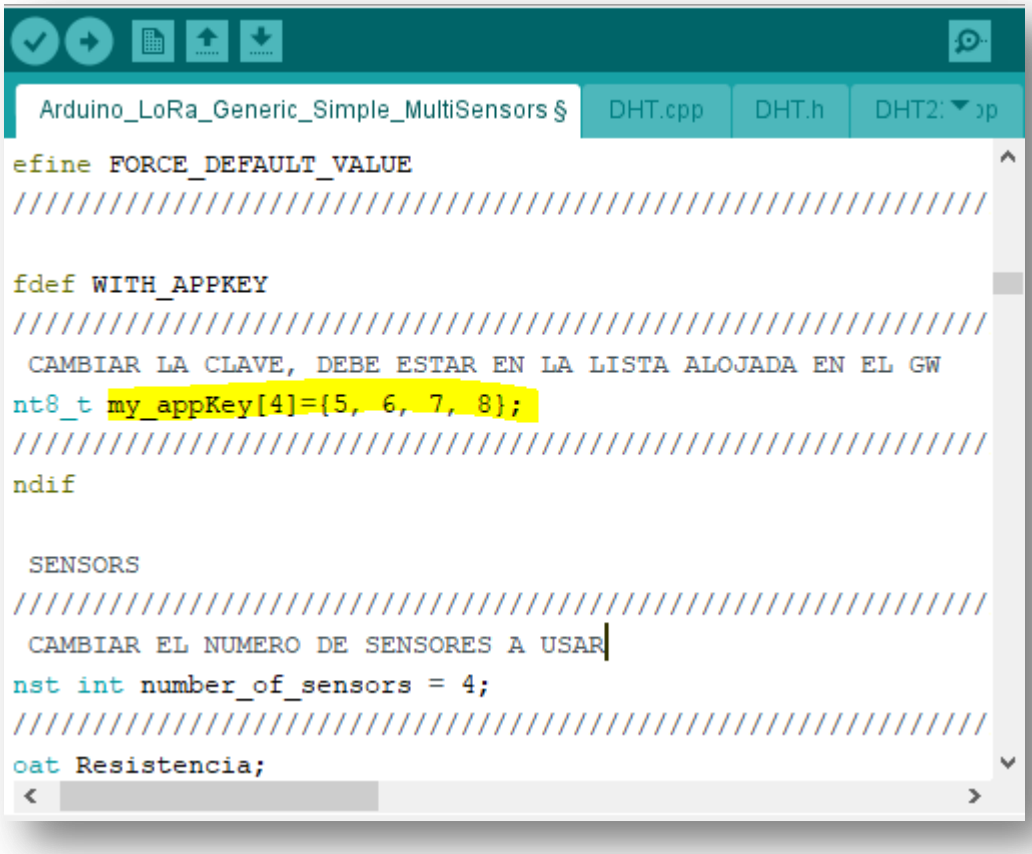


Figura 60 - Cambiar clave de aplicación

Ya con todos los cambios realizados se procede a cargar el código en la placa arduino. Se conecta el arduino a la computadora mediante el cable USB y se presiona el botón de carga, el cual está representado con una flecha direccionada hacia la derecha en la parte superior izquierda de la pantalla.

Anexo 3 Instalaciones y configuraciones del servidor

Si se está en entornos Windows, se puede instalar el servidor apache y MySQL mediante el aplicativo XAMPP[24] o WAMP[25]. La instalación es sencilla por eso no se detalla en el presente documento.

Para acceder al código fuente es deseable tener instalado GIT[26].

Ya con todo el software anteriormente mencionado, se procede a descargar el código fuente tanto del backend (API REST)[27], como del frontend[28] del sistema web. Se debe acceder mediante consola a la ruta *C:/xampp/httdocs/* aquí se debe usar el comando *git clone https://github.com/matiast02/tesis_frontend.git* el cual automáticamente copia todos los archivos del frontend. De la misma manera se debe clonar el repositorio del backend con el comando git clone.

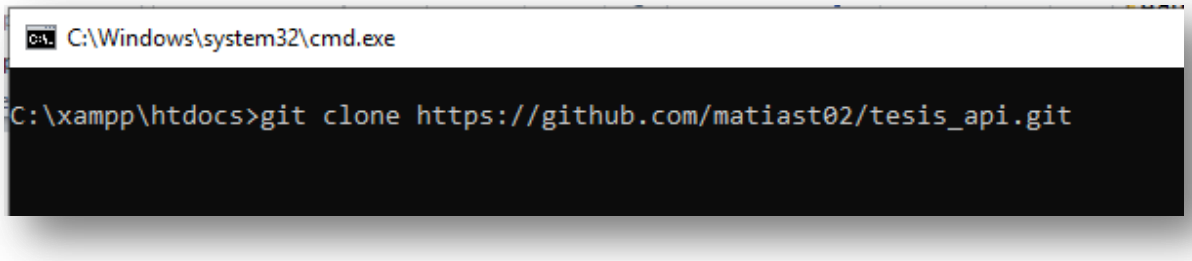


Figura 61 - Clonando repositorio tesis_api

Es momento de configurar la base de datos. Mediante el navegador web nos dirigimos a *http://localhost/phpmyadmin* y procedemos a crear la base de datos empleando el nombre que se desee.

[24] https://www.apachefriends.org/es/download.html
[25] http://www.wampserver.com/en/
[26] http://msysgit.github.com/
[27] https://github.com/matiast02/tesis_api.git
[28] *https://github.com/matiast02/tesis_fronted.git*

Figura 62 - Crear nueva base de datos mysql

Una vez creada, procedemos a importar los datos que se encuentran en *tesis.sql*, archivo que se encuentra dentro de la carpeta *tesis_api* ya clonada previamente.

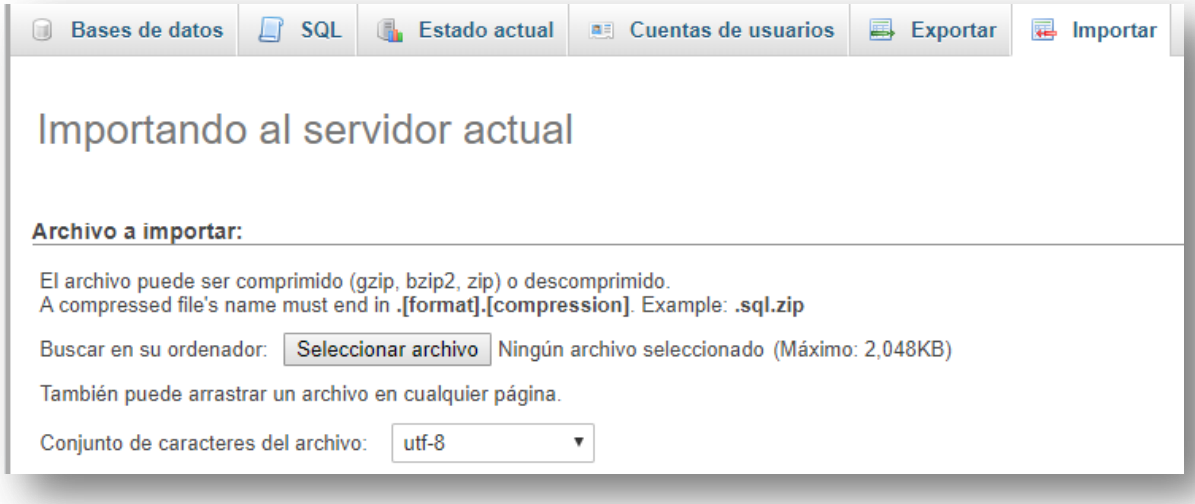


Figura 63 - Importando la base de datos

Por último se tiene que modificar el archivo *.env.example* que está dentro de la carpeta *tesis_api.* Se lo debe renombrar a *.env* y se debe modificar el contenido de este, cambiando los parámetros que figuran resaltados en la siguiente imagen.

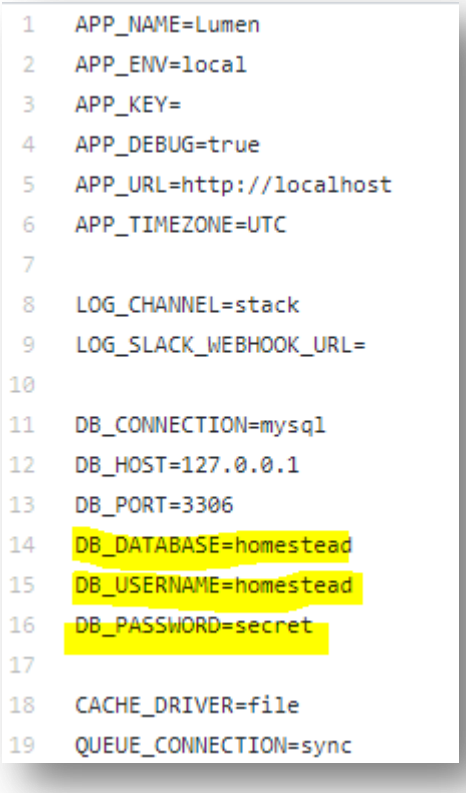


Figura 64 - Configuración de archivo .env

DB_DATABASE = nombre de la base de de datos que se creó anteriormente

DB_USERNAME = por defecto es root (si se cambio debe indicarse cual es)

DB_PASSWORD = por defecto es en blanco (no hay que escribir nada, es altamente recomendable indicar una contraseña fuerte)

Luego de configurar el archivo .ENV se debe inicializar el servicio de Apache y MySql ejecutando el panel de XAMPP y presionando el botón start de cada servicio.

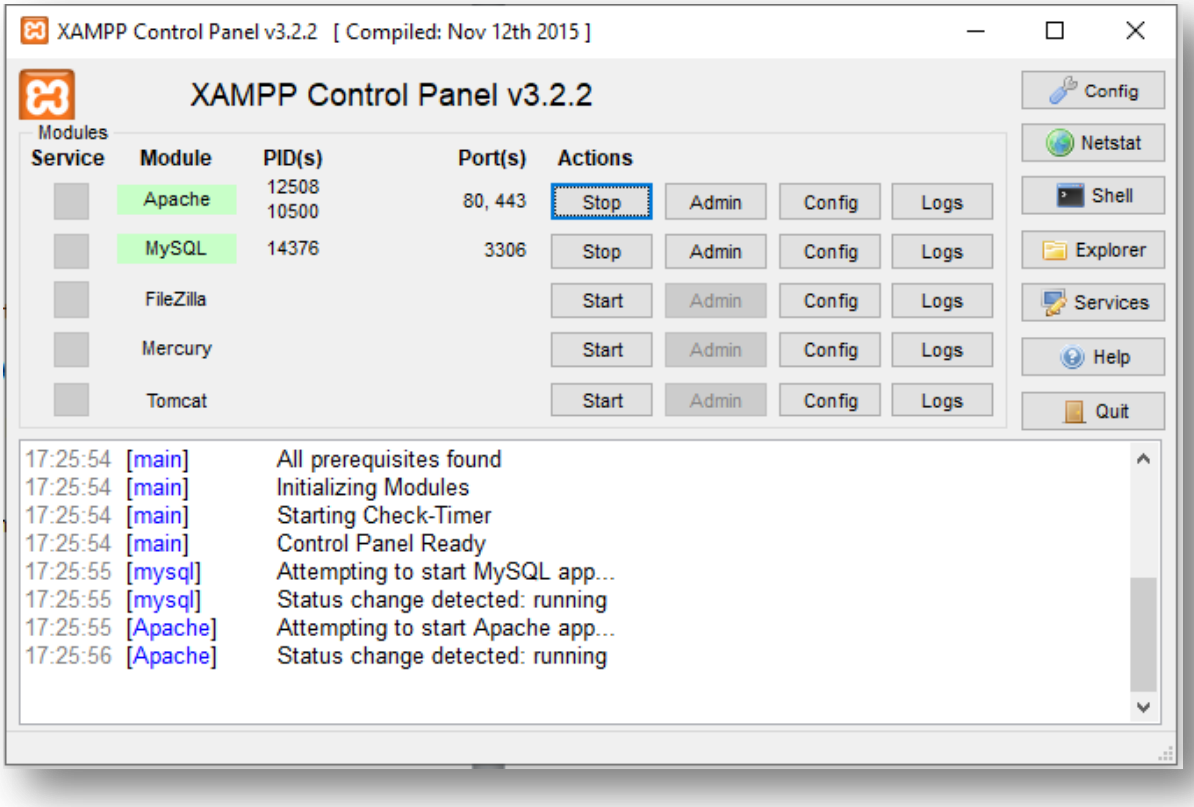


Figura 65 - panel de control xampp

Hecho todo esto, ya es posible usar el sistema, solo se debe acceder a la dirección *http://localhost/tesis_frontend/login para acceder al sistema.*

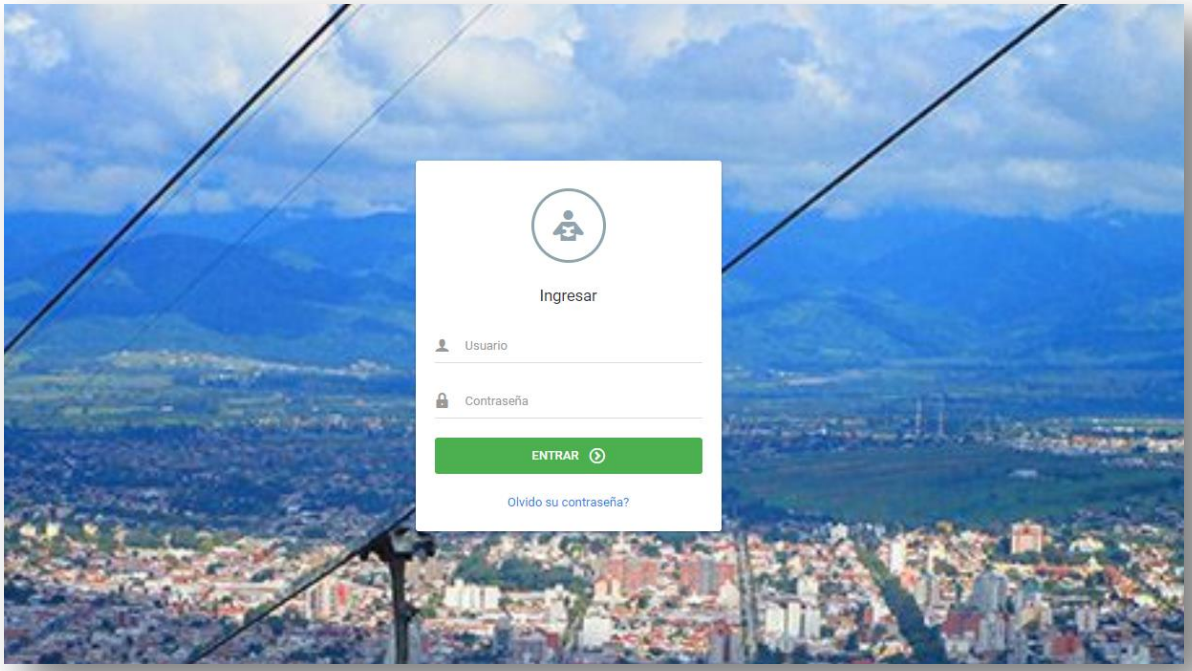


Figura 66 - Pantalla de acceso

Una vez ingresado es re direccionado a la página principal donde se visualiza la pantalla de la figura 67.

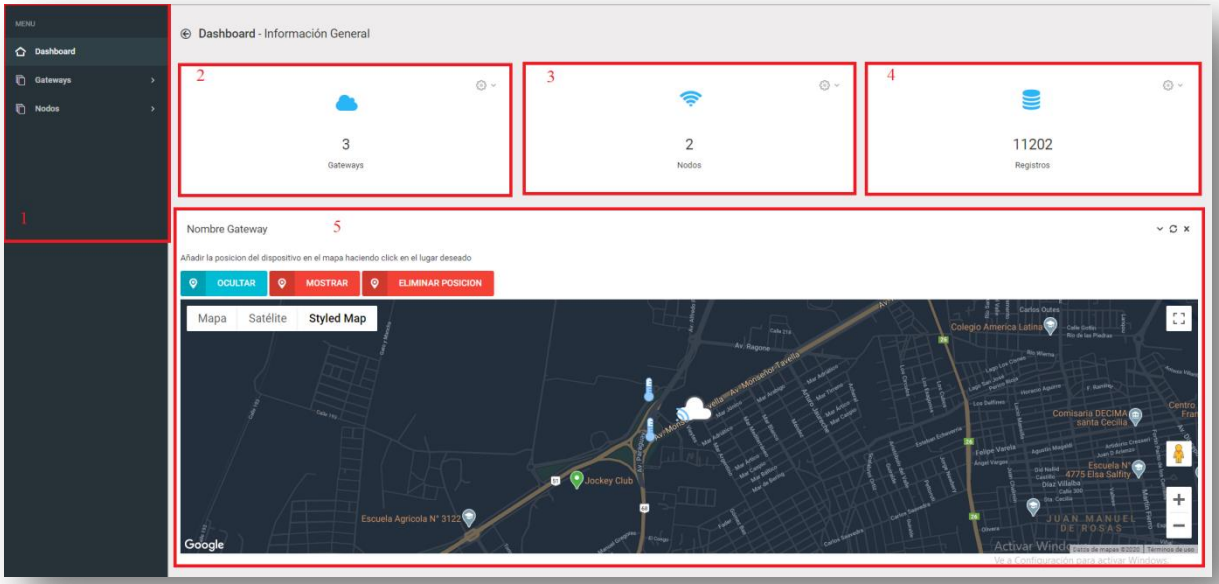


Figura 67 - Página principal

En la página principal de lado izquierdo se encuentra el menú (1), donde se puede acceder a la lista de nodos y gateways disponibles. Mientras que en la parte central superior se encuentran tres secciones, donde (2) indica la cantidad de gateways actuales en el sistema, (3) la cantidad de nodos en funcionamiento, (4) el total de mediciones realizadas hasta el momento. Por ultimo en la parte inferior se muestra un mapa con los nodos y gateways que forman parte del sistema, cada uno se diferencia por un icono, el termómetro representa los nodos sensores mientras que las nubes son los gateways.

Al hacer click en algunos de los iconos, se muestran las operaciones disponibles para dicho marcador.

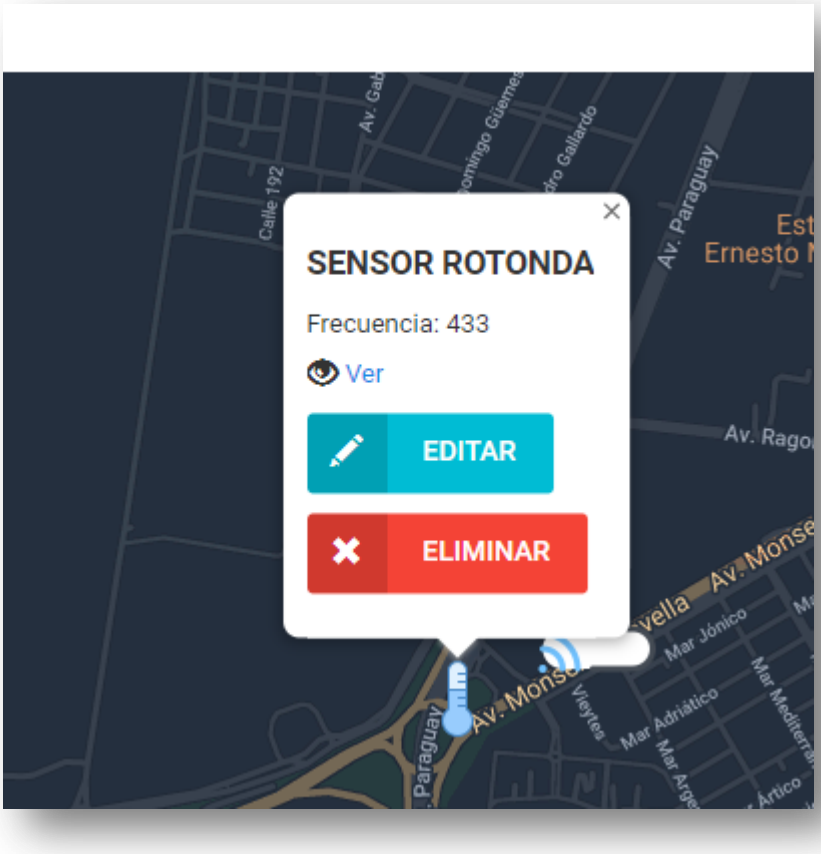


Figura 68 - operaciones de marcador

Si se escoge la opción de editar los datos, aparece un formulario flotante con los datos del nodo o Gateway para ser editados.

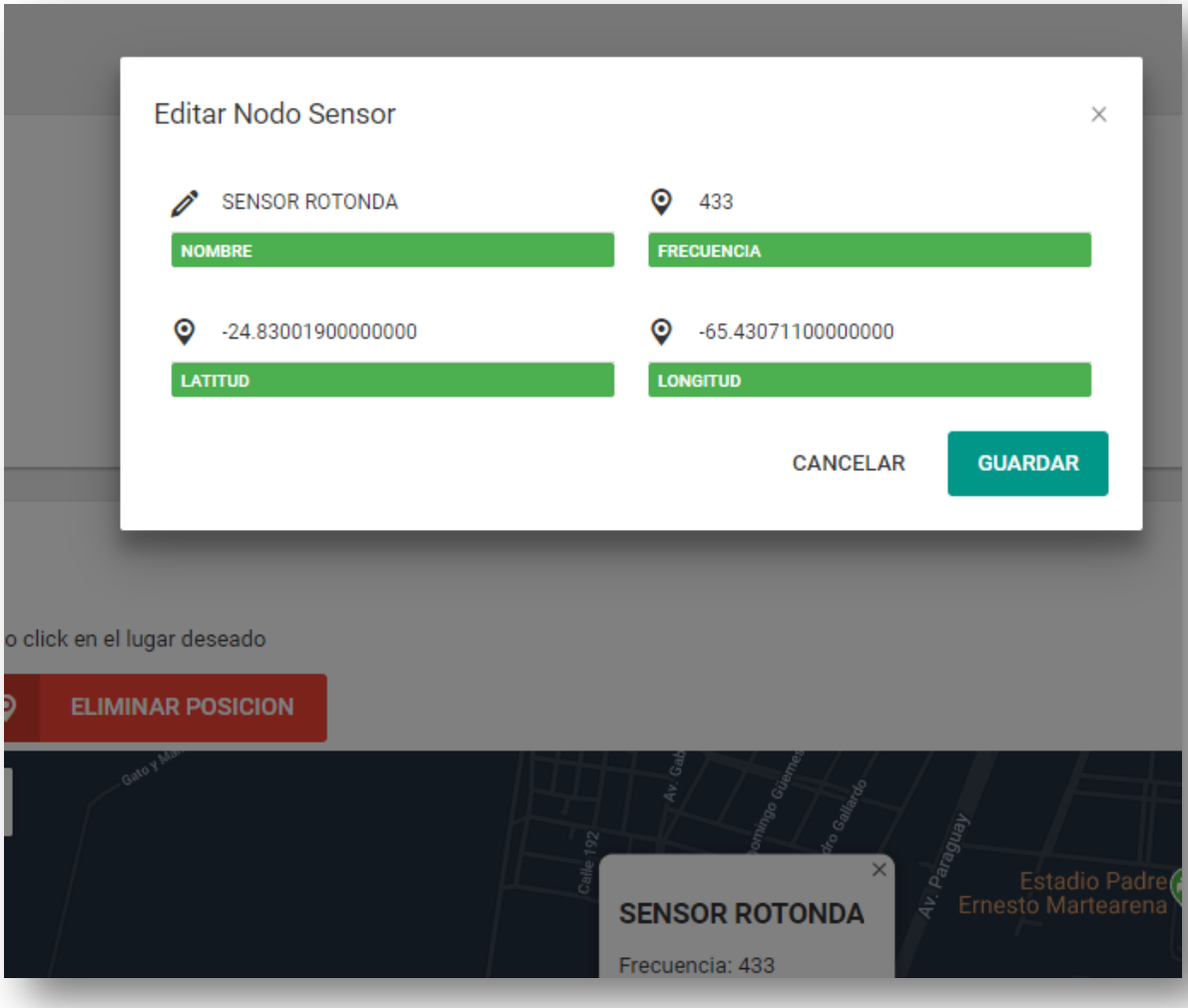


Figura 69 - formulario de edición

Si se escoge la opción ver, entonces se redirección a una página que muestra los detalles del nodo sensor.

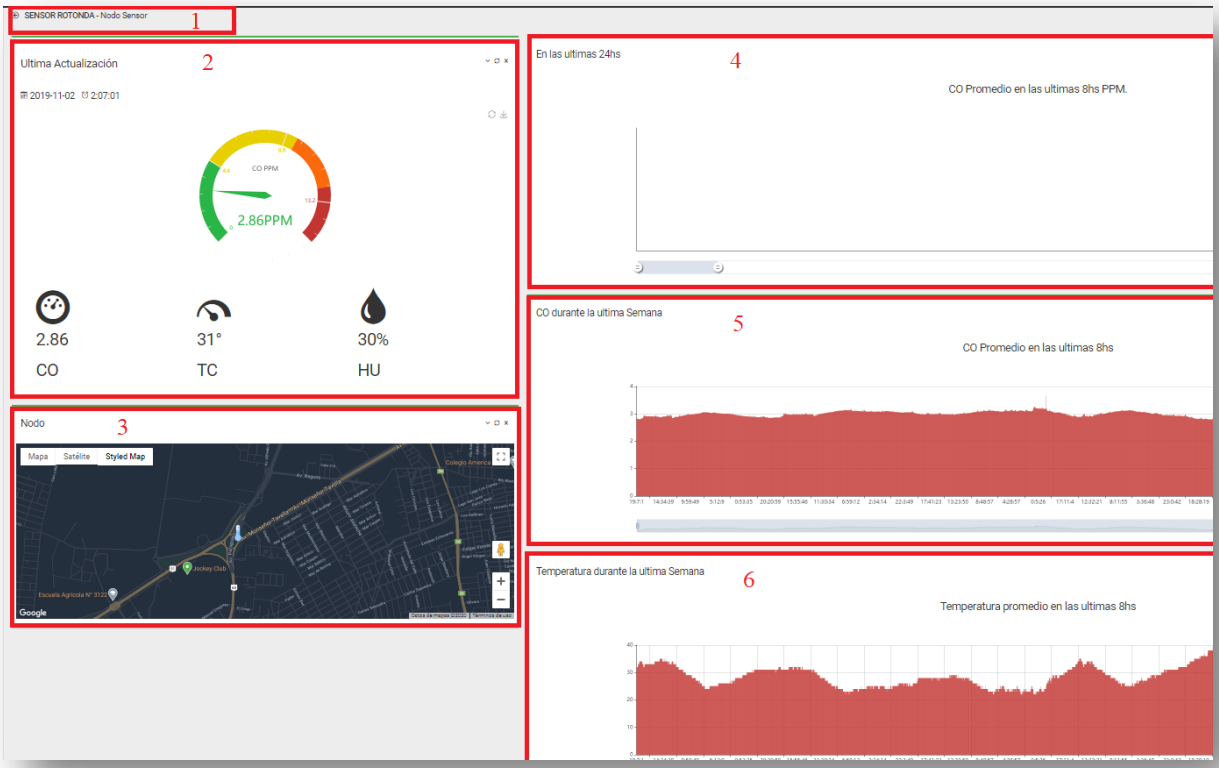


Figura 70 - Detalle de nodo sensor

Esta pantalla provee información concreta del sensor como ser:

1- El nombre del sensor
2- La última medición realizada, donde de manera gráfica expresa el valor de CO censado
3- El mapa que indica su geo posicionamiento
4- El nivel de CO en ppm durante las últimas 8 horas
5- El promedio de CO en ppm durante las últimas 8 horas
6- La temperatura en grados centígrados en las últimas 8 horas

Cada grafica cuenta con un control de desplazamiento en la parte inferior para poder visualizar los datos en diferentes momentos (horas, días).

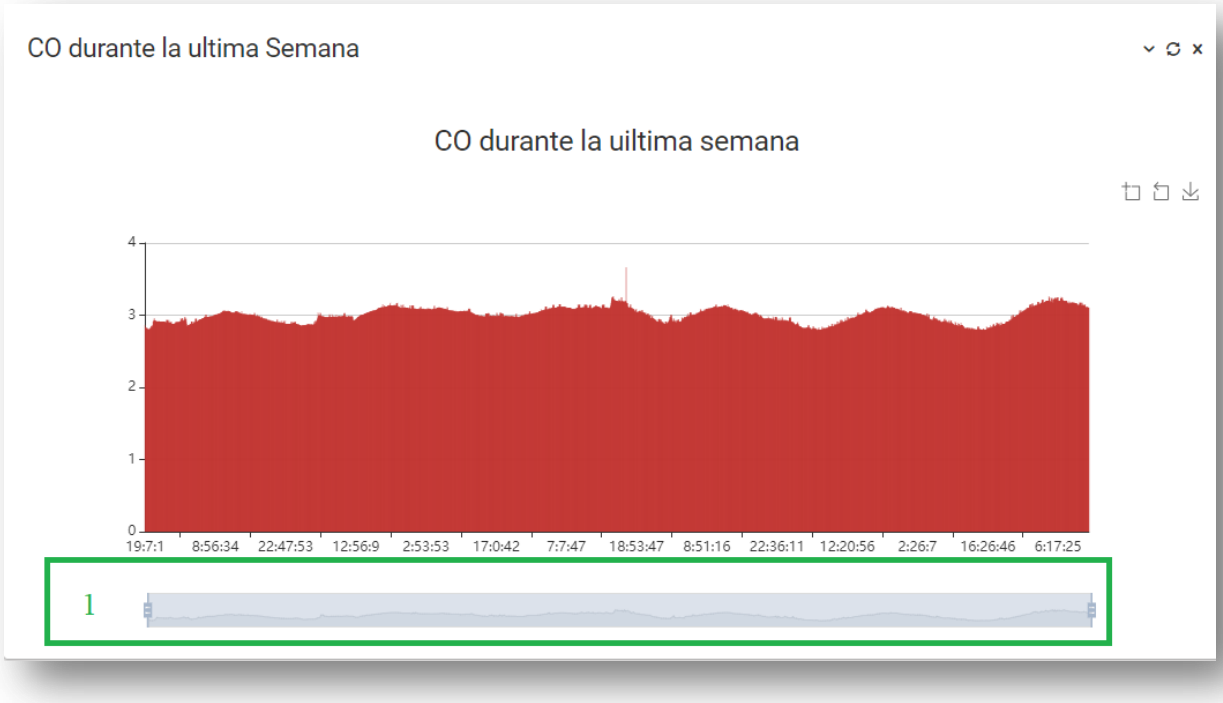


Figura 71 - control de desplazamiento

También es posible administrar los nodos y gateways desde el menú principal haciendo click en la opción lista.

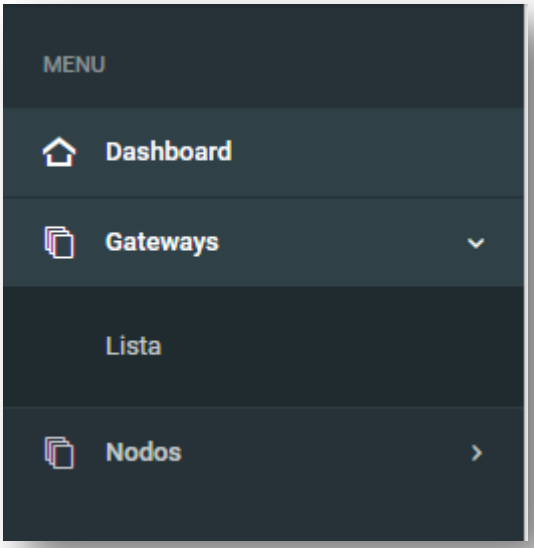


Figura 72 - Menú de opciones - >lista

Una vez seleccionada la opción lista, se muestra la pantalla de la figura 73, donde se listan cada uno de los nodos o gateways y se dispone de las opciones de edición, visualización y eliminación de los mismos.

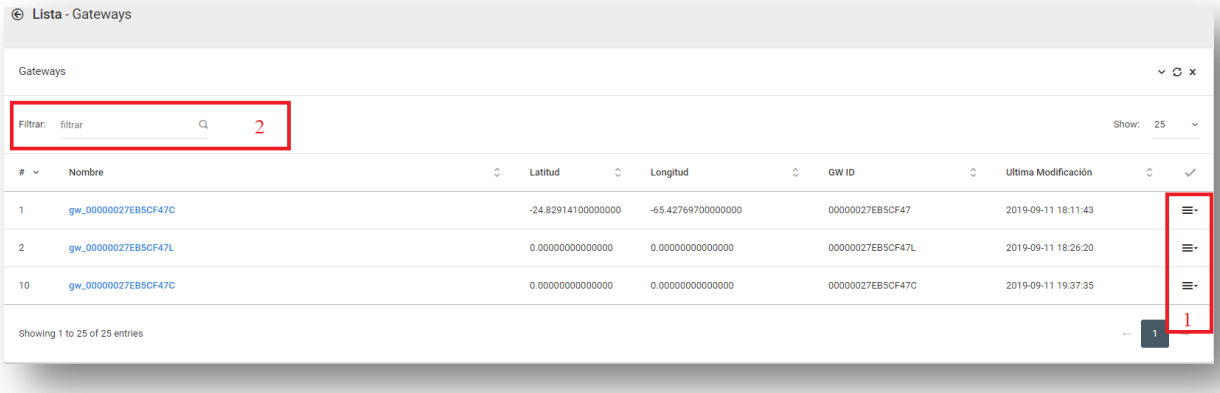


Figura 73 - Listado de gateways

Del lado derecho (1) se encuentra un icono de menú, que al presionarlo muestra las opciones disponibles que se mencionaron anteriormente. En (2) parte superior izquierda se cuenta con una caja de búsqueda para facilitar en caso que sean muchos registros.

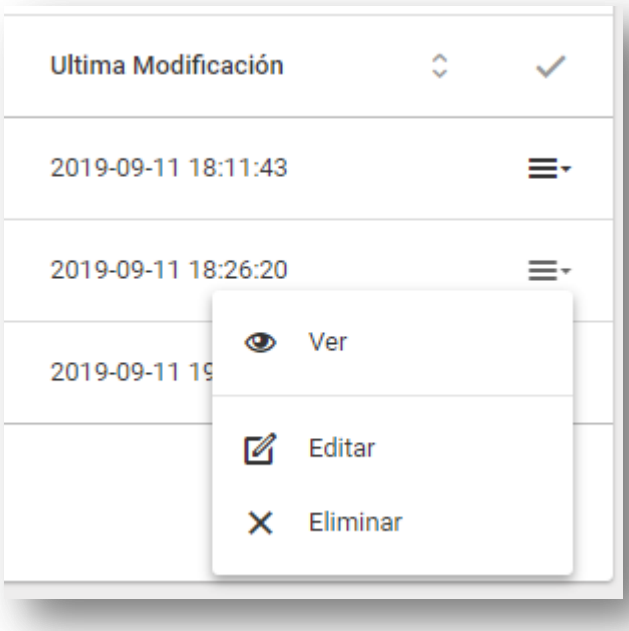


Figura 74 - Menú de opciones