

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Facultad de Ciencias e Ingeniería



**DISEÑO DE UN SISTEMA DE MONITOREO DE CONTAMINACIÓN
ACÚSTICA URBANA BAJO UNA PLATAFORMA IOT**

**TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO MECATRÓNICO**

AUTOR:

Renzo Mitsuo Murata Arca

ASESOR:

Santos Lopez, Felix Melchor

CO-ASESOR:

Portella Delgado, Jhon Manuel

Lima, 6 de Septiembre de 2020

RESUMEN

La presente tesis expone el diseño de un sistema ciberfísico de monitoreo de contaminación acústica urbana, el mismo que, combinando radiofrecuencia y conexión a internet a través de wifi, podría ser implementado en las zonas de mayor concurrencia de personas en la ciudad de Lima y tendría como principal objetivo el informar y advertir a los ciudadanos y entidades gubernamentales cuando los niveles de ruido presentes pueden afectar la salud. Además, la información recolectada permitiría a los municipios responsables del control de este problema poder tener una visión más acertada de los causantes de los mismos.

Este trabajo se dividirá en una parte física que consistirá en el diseño de un dispositivo de sensado de contaminación acústica que pueda operar de forma continua en distintas zonas de la capital siendo capaz de trabajar en la intemperie enviando información relevante de forma inalámbrica a un receptor; y una parte ciber que seguirá una arquitectura IoT con Cloud Computing que se encargará del procesamiento de los datos y su transmisión a través de internet. Este sistema será fácilmente escalable y podría ser aplicado en proyectos del Estado enfocados en la reducción de la contaminación ambiental y la concientización de la población sobre el mismo.

Dedico esta tesis a mis padres Cesar Murata y Rosario Arca por siempre brindarme su apoyo durante toda mi carrera universitaria, aconsejándome y motivándome a seguir creciendo personal y profesionalmente; a mis amigos y familia por animarme y aconsejarme siempre que lo necesité; a mis asesores Félix Santos y Jhon Portella por compartirme sus conocimientos y experiencia las cuales me ayudaron con el desarrollo de esta tesis.

CONTENIDO

1. CAPÍTULO I: INTRODUCCIÓN	1
2. CAPÍTULO II: MARCO TEÓRICO Y ESTADO DE LA TECNOLOGÍA	4
2.1. EL RUIDO:	4
2.1.1 Nivel de presión sonora (SPL):	4
2.1.2. Ponderación acústica:	5
2.2. INTERNET DE LAS COSAS (IoT):	6
2.2.1. Tipos de redes IoT:	6
2.2.2. Topologías de red.....	7
2.2.3. Arquitecturas IoT:	9
2.3 DISEÑO DE LA ARQUITECTURA DEL SOFTWARE	10
2.3.1. Attribute-Driven Design	10
2.4. PRODUCTOS COMERCIALES:	11
2.4.1 IN_NOISE:	11
2.4.2. KUNAK NOISE N10:	11
2.4.3. Sensor de ruido TA120:	12
2.5. TESIS Y DOCUMENTOS:	13
2.5.1. The Implementation of Low-cost Urban Acoustic Monitoring Devices.....	13
2.5.2. Desarrollo de un sistema de medición de ruido acústico	13
2.5.3. Diseño e implementación de una red inalámbrica de sensores inteligentes para el registro de la contaminación acústica.....	14
3. CAPÍTULO III: DISEÑO CONCEPTUAL DEL SISTEMA	15
3.1. SELECCIÓN DE RED IOT PARA SISTEMA CIBER	15
3.1.1. Topología LoRaWAN:.....	16
3.2. DISEÑO CONCEPTUAL DEL SISTEMA FÍSICO.....	16
3.2.1 Lista de requerimientos	16
3.2.2. Estructura de funciones.....	19
3.2.3. Matriz morfológica	23
3.2.4. Conceptos de solución	27
3.2.5. Evaluación Técnico-Económica.....	32
3.3. DISEÑO CONCEPTUAL DEL SISTEMA CIBER	34
3.3.1 Lista de requerimientos	34
4. CAPÍTULO IV: DISEÑO DEL SISTEMA CIBERFÍSICO INTEGRADO	35
4.1. FUNCIONAMIENTO DEL SISTEMA INTEGRADO	35
4.2. DISEÑO DEL SISTEMA FÍSICO	36
4.2.1 Arquitectura del hardware.....	36
4.2.2 Modelo 3D del sistema	37
4.2.3. Selección de componentes electrónicos y de procesamiento	38
4.2.3.1. Nodo Sensor:.....	38
4.2.3.2. Gateway	43
4.2.4. Selección de componentes de energía y potencia	45
4.2.4.1. Cálculos previos.....	45
4.2.4.2. Componentes de potencia	46
4.2.4.3. Componentes de energía	47

4.2.5. Selección de componentes mecánicos	51
4.2.5.1. Protección y sujeción	51
4.2.5.2. Protección de micrófono	57
4.2.6. Diagramas de conexiones:	57
4.2.7. Diagramas de flujo:	59
4.2.7.1. Diagrama general de nodo sensor	59
4.2.7.2. Configuración de inicio.....	60
4.2.7.3. Verificar estado de sensor de sonido.....	61
4.2.7.4. Verificar alimentación del nodo sensor.....	62
4.2.7.5. Revisar validez de data	63
4.2.7.6. Medición y validación de data	64
4.3. DISEÑO DE ARQUITECTURA DEL SISTEMA CIBER:	64
4.3.1. Procesos de diseño ADD 3.0.....	65
4.3.1.1. Revisión de entradas (Paso 1)	65
4.3.1.2. Iteración 1: Selección de arquitectura (Pasos 2 al 7)	67
4.3.1.3. Iteración 2: Flujo de datos y monitoreo (Pasos 2 al 7).....	73
4.3.1.4. Iteración 3: Selección de Tecnologías (Pasos 2 al 7).....	76
5. CAPÍTULO V: PROTOTIPO Y PRUEBAS DEL SISTEMA.....	82
5.1. PROTOTIPO GATEWAY	82
5.2. PROTOTIPO NODO SENSOR.....	84
5.3. PRUEBAS DE CONECTIVIDAD DEL SISTEMA	87
5.4. PRUEBAS DEL SENSOR DE SONIDO.....	87
6. CAPÍTULO VI: COSTOS DEL SISTEMA	91
6.1. COSTO DEL SISTEMA FÍSICO	91
6.1.1. Componentes eléctricos y electrónicos	91
6.1.2. Componentes Mecánicos	92
6.1.2.1. Componentes.....	92
6.1.2.1. Mecanizado	92
6.1.3. Caso de estudio para implementación en campo del sistema diseñado	93
6.2. COSTO DEL SISTEMA CIBER.....	94
6.2.1 AWS IoT	95
6.2.2 DynamoDB	98
6.2.3 Amazon S3.....	99
6.2.4 AWS Lambda.....	99
6.2.5 AWS CloudFront	100
6.2.6 API Gateway	101
6.2.7 Amazon Cognito	101
6.2.8 Amazon CloudWatch.....	101
6.3. COSTO TOTAL DEL SISTEMA.....	102
CONCLUSIONES Y RECOMENDACIONES.....	103
BILIOGRAFIA.....	105
ANEXOS Y PLANOS.....	108

ÍNDICE DE TABLAS

Tabla 2.1 Lista de requerimientos sistema físico	17
Tabla 3.1 Matriz morfológica Nodo Sensor.....	24
Tabla 3.2 Matriz morfológica Gateway	26
Tabla 3.3 Valor técnico del sistema	32
Tabla 3.4 Valor económico del sistema	32
Tabla 3.5 Lista de requerimientos sistema ciber.....	34
Tabla 4.1 Sensores de sonido.....	39
Tabla 4.2 Sensores de lluvia	39
Tabla 4.3 Sensor de corriente.....	40
Tabla 4.4 Microcontrolador Nodo Sensor.....	41
Tabla 4.6 Módulos buzzer.....	41
Tabla 4.7 Transceptores LoRa	42
Tabla 4.8 Antena Nodo Sensor	43
Tabla 4.5 Microcontrolador Gateway	43
Tabla 4.9 Modulo RAK2245	44
Tabla 4.10 Antena Nodo Sensor	45
Tabla 4.11 Consumo máximo de potencia de componentes seleccionados.....	45
Tabla 4.12 Selección de placa UPS	46
Tabla 4.13 Protección contra sobrecarga	47
Tabla 4.14 Fuente de alimentación Nodo Sensor.....	48
Tabla 4.15 Batería de ion de litio Nodo Sensor	49
Tabla 4.16 Fuente de alimentación Gateway	50
Tabla 4.17 Batería de ion de litio Gateway.....	51
Tabla 4.18 Caja rittal Nodo Sensor.....	52
Tabla 4.19 Elemento de sujeción Nodo Sensor	53
Tabla 4.20 Masa y peso de los componentes seleccionados.....	53
Tabla 4.21 Parámetros para velocidad de diseño.....	53
Tabla 4.22 Caja Rittal Gateway	56
Tabla 4.23 Elemento de sujeción Gateway	57
Tabla 4.24 Elemento protector de micrófono	57
Tabla 4.25 Descripción de objetivos de diseño.....	65
Tabla 4.26 Escenarios de atributos de calidad	66

Tabla 4.27 Restricciones del proyecto	67
Tabla 4.28 Preocupaciones del proyecto.....	67
Tabla 4.29 Decisiones de diseño – paso 4	69
Tabla 4.30 Decisiones de diseño – paso 5	70
Tabla 4.31 Descripción de subsistemas	71
Tabla 4.32 Revisión de objetivos.....	72
Tabla 4.33 Decisiones de diseño – paso 4	73
Tabla 4.34 Decisiones de diseño – paso 5	74
Tabla 4.35 Revisión de objetivos.....	76
Tabla 4.36 Decisiones de diseño – paso 4	77
Tabla 4.37 Decisiones de diseño – paso 5	78
Tabla 4.38 Revisión de objetivos.....	80
Tabla 5.1 Resultados de pruebas del sensor de sonido	89
Tabla 6.1 Costo de componentes eléctricos y electrónicos Nodo Sensor	91
Tabla 6.2 Costo de componentes eléctricos y electrónicos Gateway.....	91
Tabla 6.3 Costo de componentes mecánicos Nodo Sensor.....	92
Tabla 6.4 Costo de componentes mecánicos Gateway	92
Tabla 6.5 Costo del mecanizado Nodo Sensor	92
Tabla 6.6 Costo del mecanizado Gateway	93
Tabla 6.7 Costo anual del sistema ciber.....	102

ÍNDICE DE FIGURAS

Figura 1.1 Relación entre tipo de ambiente y nivel de decibeles	1
Figura 2.1 Curvas de ponderación acústica	5
Figura 2.2 Capacidades de distintas redes IoT	7
Figura 2.3 Topología estrella	8
Figura 2.4 Topología malla.....	8
Figura 2.5 The 4 stage IoT solution architecture	9
Figura 2.6 Proceso ADD 3.0.....	10
Figura 2.7 Dispositivo IN_NOISE.....	11
Figura 2.8 Dispositivo Noise N10.....	12
Figura 2.9 Dispositivo Ta120	12
Figura 2.10 Prototipo de sensor con dispositivo Tronsmart MK908ii.....	13
Figura 2.11 Componentes del sistema hardware	14
Figura 2.12 Red inalámbrica de sensores.....	14
Figura 3.1 Topología LoRaWAN	16
Figura 3.2 Bloque energía.....	20
Figura 3.3 Bloque potencia	20
Figura 3.4 Bloque sensores	21
Figura 3.5 Bloque electrónica	21
Figura 3.6 Bloque control	22
Figura 3.7 Bloque comunicación	22
Figura 3.9 Bloque mecánica	23
Figura 3.10 Diseño conceptual solución 1	28
Figura 3.11 Diseño conceptual solución 2	30
Figura 3.12 Diseño conceptual solución 3	31
Figura 3.13 Evaluación técnico económica	33
Figura 4.1 Arquitectura de hardware nodo sensor	36
Figura 4.2 Arquitectura de hardware gateway	37
Figura 4.3 Modelo 3D Nodo Sensor	37
Figura 4.4 Modelo 3d Gateway	38
Figura 4.5 Diagrama de distribución de componentes Nodo Sensor.....	52
Figura 4.6 Diagrama de fuerzas del Nodo Sensor	55
Figura 4.7 Diagrama de distribución de componentes Gateway.....	56

Figura 4.8 Diagrama de conexiones Gateway	58
Figura 4.9 Diagrama de conexiones Nodo Sensor.....	58
Figura 4.10 Diagrama de flujo general	60
Figura 4.11 Diagrama de flujo de configuración inicial	60
Figura 4.12 Diagrama de flujo de verificación de estado de sensor	61
Figura 4.13 Diagrama de flujo de verificación de alimentación NS	62
Figura 4.14 Diagrama de flujo de validez de data	63
Figura 4.15 Diagrama de flujo de medición y validación.....	64
Figura 4.16 Arquitectura Microsoft Azure IoT.....	70
Figura 4.17 Arquitectura inicial.....	71
Figura 4.18 Modelo de arquitectura sateless basada en Azure	74
Figura 4.19 Arquitectura iteración 2.....	75
Figura 4.20 Arquitectura iteración 2.....	80
Figura 5.1 Prototipo Gateway	82
Figura 5.2 Programa principal Gateway	83
Figura 6.3 Registro de Gateway en TTN.....	83
Figura 5.4 Prototipo Nodo Sensor	84
Figura 5.5 Registro de sensor en aplicación	85
Figura 5.6 Código 1 Nodo Sensor.....	85
Figura 5.7 Código 2 Nodo Sensor.....	86
Figura 5.8 Decodificador en TTN.....	86
Figura 5.9 Data recibida por la aplicación en TTN.....	87
Figura 5.10 Sonómetro AWA5661	87
Figura 5.11 Materiales para las pruebas.....	88
Figura 5.12 Medición de sensor profesional.....	88
Figura 5.13 Mediciones del sensor de sonido.....	89
Figura 6.1 Cobertura de Gateways.....	93
Figura 6.2 Distribución de Nodos Sensores.....	94
Figura 6.3 Costos por conectividad AWS.....	95
Figura 6.4 Costos por mensajería AWS.....	95
Figura 5.5 Costos por registro y sombra AWS	96
Figura 6.6 Costos por motor de reglas AWS	97
Figura 6.7 Costos de DynamoDB	98
Figura 6.8 Costos de Amazon S3.....	99

Figura 6.9 Costos de Amazon Cloudfront	100
Figura 6.10 Costos de API Gateway	101
Figura 6.11 Capa gratuita Amazon Cloudwatch	101

1. CAPÍTULO I: INTRODUCCIÓN

Actualmente, la contaminación sonora urbana es uno de los principales problemas que afecta a la población de las grandes ciudades; esto se debe a que la exposición de las personas a niveles altos de ruido puede producir presión alta, insomnio, vértigo, estrés, pérdida de la audición y dificultades al hablar [1]. Según un análisis de contaminación urbana en zonas rurales y urbanas, los niveles de contaminación en zonas urbanas suelen ser 20 decibeles superior al de las zonas rurales [2]. En la figura 1.1 se muestran los niveles de presión sonora típicos en distintos tipos de ambientes. Como se puede apreciar, la gravedad de los daños a la salud aumenta mientras mayor sean los decibeles registrados; una persona realizando actividades rutinarias no debería estar expuesto a más de 70 decibeles y se recomienda usar tapones u orejeras cuando se superan los 85 decibeles.



Figura 1.1 Relación entre tipo de ambiente y nivel de decibeles

Fuente: [1]

Según la investigación acerca del impacto de la contaminación acústica en la salud pública que realizó Guisela Yábar Torres, doctora en medio ambiente y desarrollo sostenible, se concluyó que el tráfico vehicular es la principal fuente generadora de ruido, dejando en segundo plano a las empresas, construcciones e industrias [3]. Además, una evaluación de contaminación acústica en zonas de tráfico realizada en la ciudad de

Curitiba en Brasil determino que reduciendo el 50% de tráfico vehicular se podría reducir hasta 3 decibeles los niveles de contaminación acústica de la ciudad [4]. Una forma de proteger nuestra salud frente a este problema es evitar la exposición prolongada a los ruidos y/o el uso de protectores auditivos [5]. Si bien se puede estimar cuando un ruido llega a ser molesto, un conocimiento correcto de los niveles de ruido de la zona donde nos ubicamos nos puede ayudar a tomar las medidas adecuadas.

Hoy en día los Smartphone poseen la capacidad de medir el nivel de contaminación acústica de un lugar a través de aplicaciones sin embargo muchas veces el micrófono del celular no permite un buen análisis [6]. Además, difícilmente una persona se pondrá a medir los niveles de contaminación desde su celular arriesgándose a robos y, considerando que un buen registro requiere una medición constante [7] y no una instantánea, la medición realizada por el ciudadano mismo no es una buena opción. Por parte del estado existen procedimientos para el correcto sensado de los niveles acústicos de una zona, sin embargo, estos trabajos requieren la presencia de trabajadores y el uso de equipos especiales que deben ser colocados de forma temporal en determinadas zonas de la ciudad. Estos estudios son realizados de forma poco frecuente por lo que a medida que haya pasado mayor tiempo entre un sensado y otro los resultados comenzarán a ser más erróneos [7].

En busca de una solución a este problema, se plantea una tesis que consiste en diseñar un sistema de monitoreo automatizado de contaminación acústica urbana de bajo costo, bajo consumo energético, que pueda ser implementado en zonas estratégicas de la ciudad de Lima y funcionen bajo una arquitectura IoT, la cual permitirá un sensado constante sin la necesidad de movilizar costosos equipos y/o personal. Este sistema permitirá que los ciudadanos sean conscientes de los niveles de ruido presentes en la zona donde se ubiquen ya que toda la información recolectada por el sistema será almacenada en una base de datos la cual podrá ser consultada por usuarios externos a través de una app; también le permitirá a las municipalidades tener un mejor control sobre las zonas críticas en cuestión a contaminación sonora pudiendo saber cuándo una zona empieza a tener altos niveles de ruido sin necesidad de esperar las quejas de sus ciudadanos.

Objetivo general:

Diseñar un sistema ciberfísico de monitoreo de contaminación acústica urbana de bajo costo que funcione bajo una arquitectura IoT y tenga una arquitectura Cloud que brindar información actualizada a los ciudadanos a través de aplicativos móviles y permita a las municipalidades almacenar data para identificar puntos críticos a través de estudios más especializados.

Objetivos específicos

- Realizar un estudio bibliográfico acerca de la problemática y la forma en que operan los sistemas de monitoreo acústico, con la finalidad de sustentar la investigación
- Realizar un estudio del estado de arte actual que permita identificar las tecnologías existentes similares que permitan dar un enfoque más realista y acotado a la investigación.
- Definir los requerimientos y especificaciones técnicas del sistema físico y ciber para que cumplan con las leyes y normas que se aplican el campo de la medición acústica.
- Elaborar el diseño conceptual del sistema físico, en los dominios mecánico y electrónico, teniendo en consideración aspectos técnicos y económicos.
- Elaborar el diseño conceptual del sistema ciber considerando aspectos: funcionales, comerciales, de seguridad, de tiempo de respuesta y de calidad de datos.
- Desarrollar el diseño del sistema físico mediante cálculos para el diseño mecánico y electrónico del sistema.
- Diseñar la arquitectura IoT del sistema ciber siguiendo el proceso de diseño “Attribute-Driven Design”.
- Implementar un prototipo del sistema y realizar pruebas de comunicación por radiofrecuencia, conexión al servidor web y medición de contaminación sonora con el prototipo.
- Estimar los costos totales del sistema ciberfísico en una posible implementación, considerando costos de los componentes seleccionados y los costos de los servicios Cloud.

2. CAPÍTULO II: MARCO TEÓRICO Y ESTADO DE LA TECNOLOGÍA

En este capítulo se explicará la teoría detrás de la medición de contaminación acústica, las redes IoT y el diseño de arquitecturas de software. Además, se revisarán productos comerciales y trabajos de investigación sobre sistemas de monitoreo acústico para tener una orientación más clara de las funcionalidades que debe lograr el sistema a diseñar.

2.1. El Ruido:

Según el autor A.D.E Alonso [8] el ruido es una manifestación de las energías liberadas en distintas actividades cotidianas, que puede afectar el oído humano y la salud de las personas. Este también se puede describir como un sonido intempestivo y molesto que puede producir efectos psicológicos y fisiológicos que alteren a una persona o a un grupo de estas. El ruido es un tipo de sonido, y son las circunstancias subjetivas de quien lo escucha lo que determina su clasificación. Por ello, mediante procedimientos estadísticos y teniendo en cuenta algunas características de los individuos, es posible determinar el nivel de ruido aceptable en una ubicación determinada, dada ciertas circunstancias.

Para medir un ruido se consideran tres magnitudes, las cuales están relacionadas directamente con su agresividad. En primer lugar, se tiene a la intensidad, la cual es el nivel que está asociado a la cantidad de energía empleada para generar un ruido, esta se mide en decibelios (dB) y es la magnitud más importante en temas de contaminación acústica. La duración y frecuencia de exposición, también son necesarias para saber el nivel de contaminación que genera el ruido [8] por lo que ruidos con la misma intensidad son considerados más o menos dañinos dependiendo de sus otros factores.

2.1.1 Nivel de presión sonora (SPL):

Según el autor Ramos Ridao [9], en la acústica, el nivel de presión sonora (también conocido como amplitud de onda) que es la diferencia entre la presión mínima audible y la presión instantánea que generan las ondas sonoras, utiliza el decibelio. El decibelio es la décima parte de un belio y es una unidad logarítmica, matemáticamente escalar y adimensional. El Belio es el logaritmo decimal que relaciona la magnitud de interés y la

magnitud de referencia en una medición [10]. La siguiente ecuación muestra el cálculo del nivel de presión sonora:

$$L_p = 20 \log(P/P_0) \text{ dB},$$

donde:

- L_p es el nivel de presión sonora.
- P es la presión instantánea
- P_0 es la presión de referencia ($20 \mu\text{Pa}$)

Esta medida es utilizada por los sonómetros para la determinación de los niveles de contaminación acústica ya que, a una distancia suficiente de la fuente de ruido, el cuadrado de esta medida es proporcional a la intensidad del sonido.

2.1.2. Ponderación acústica:

Si bien el oído humano tiene la capacidad de percibir sonidos entre 20 Hz y 20kHz, este no posee la misma sensibilidad para todo el rango de frecuencias. Debido a eso, se han establecido y normalizado diferentes curvas de ponderación acústica las cuales varían en función de la frecuencia tratando de asemejarse lo mejor posible al oído humano. En la figura 2.1 se muestran las 3 principales curvas de ponderación que se utilizan para el análisis acústico, estas curvas tratan de replicar la respuesta del oído humano en frecuencia a distintos niveles de presión sonora siendo la ponderación A la más baja y la ponderación C la más alta.

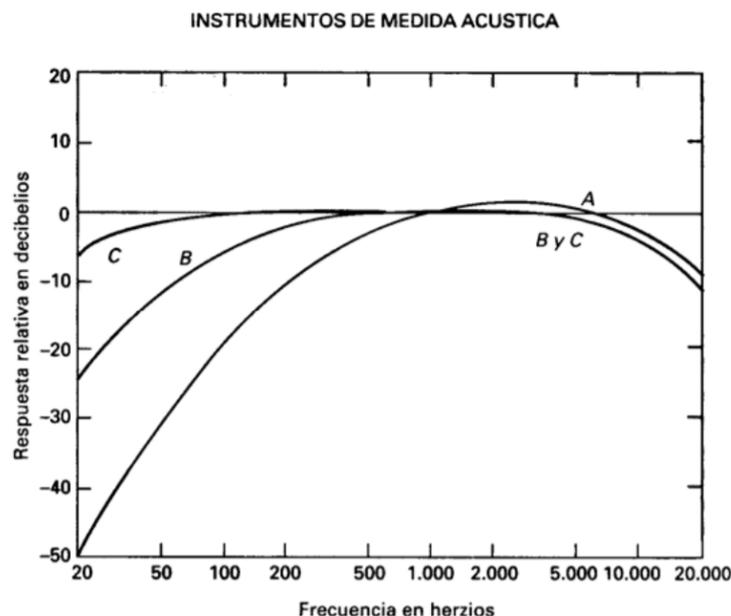


Figura 2.1 Curvas de ponderación acústica

Fuente: [9]

El nivel sonoro más utilizado para el análisis urbano de contaminación acústica tiene la Ponderación A debido a que es la que presenta un mayor cuidado con las personas frente a altos niveles de ruido. Cuando se presenta esta ponderación se suele representar el valor acompañado con el símbolo dB(A), representando así la ponderación de los niveles sonoros [9]. Esta ponderación suele estar presente en la mayoría de sonómetros integradores y es exigido en los protocolos de medición de contaminación sonora

2.2. Internet de las Cosas (IoT):

Internet de las Cosas (IoT) consiste en la interrelación inteligente de personas, procesos, datos y objetos, lo que genera que las conexiones de red sean más importantes ypreciadas que antes. Esto se produce mediante la transformación de la información en acciones que crean mejores experiencias, nuevas funcionalidades y oportunidades económicas sin precedentes para personas, países y empresas [11].

2.2.1. Tipos de redes IoT:

Lo primero que se debe seleccionar al momento de realizar un proyecto con IoT es el tipo de red que se va a utilizar. En la actualidad existen diferentes opciones para transmitir data entre sensores IoT y la Web o los aplicativos móviles, cada una presente distintas ventajas y desventajas por lo que no existe una solución universal.

Short Range Wireless Networks:

Este tipo de comunicación de corto alcance requiere del despliegue de repetidores para hacerlos viables en estructuras actuales, si bien han sido claves para el éxito inicial de IoT, su efectividad se ve bastante reducida cuando se realizan despliegues amplios de sensores. En esta categoría se encuentran tecnologías como ZigBee, Z-Wave, Bluetooth, Wifi, etc [12] [13].

M2M (Machine to Machine) – 2G, 3G, 4G & 5G networks:

En la actualidad si se requiere conocer la ubicación, el estado o directamente realizar interacción entre 2 dispositivos de forma global con cobertura geográfica nacional o incluso internacional la solución global es el uso de tarjetas Sim y redes GPRS/3G de operadores telefónicos locales. El problema con este tipo de red es su alto consumo energético, la necesidad de un pago mensual por suscripción y el alto costo de los equipos [12].

LPWAN – Low Power Wide Area Networks:

Este tipo de red nativa de IoT se ha popularizado bastante desde su lanzamiento por parte de la empresa SigFox en el 2012. Se caracteriza por su muy bajo consumo energético, su largo alcance y bajo costo de dispositivos. Actualmente existen 2 tecnologías que utilizan este sistema SigFox y LoRaWAN [13] [12].

En la figura 2.2 se puede observar una comparación de las capacidades de los tipos de red mencionados.

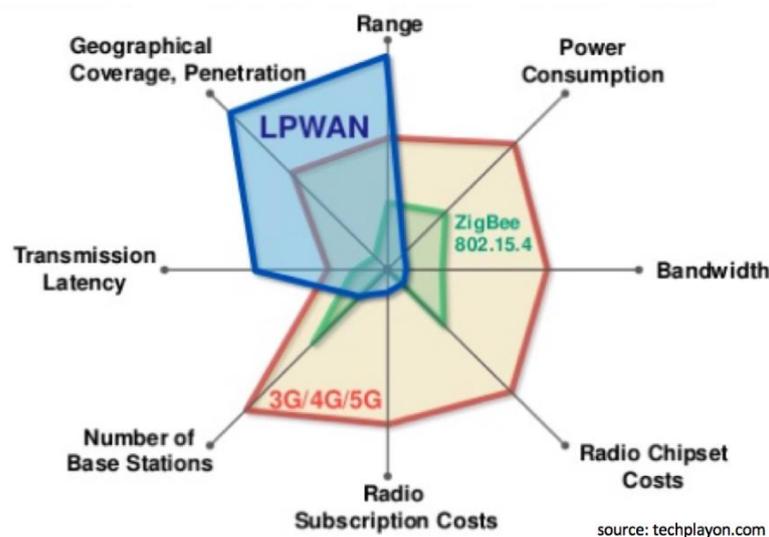


Figura 2.2 Capacidades de distintas redes IoT

Fuente: [12]

2.2.2. Topologías de red

Para poder implementar una red de sensores IoT se debe seleccionar un tipo de topología a seguir al momento de la distribución de dispositivos. Las topologías básicas utilizadas en este tipo de redes son estrella y malla. Cada topología presenta distintas ventajas y desventajas [14].

Topología en estrella:

En este tipo de red todos los sensores envían información a una puerta de enlace (Gateway) y esta lo envía a una base de datos. La información pasa una sola vez por el Gateway y este lo retransmite al exterior, los nodos sensores no tienen comunicación directa y se requiere de la intervención del Gateway para que estos se comuniquen. Esta topología tiene la ventaja de tener un menor consumo de energía sin embargo no posee

un camino alternativo de comunicación por lo que en caso falle la comunicación de un nodo la información proveniente de sus sensores asociados se perderá. En la figura 2.3 se observa un ejemplo de topología estrella donde “S” representa el nodo sensor y “G” representa el Gateway:

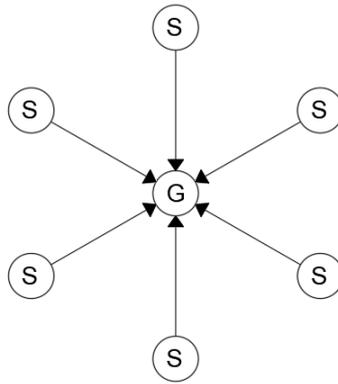


Figura 2.3 Topología estrella

Topología en malla:

Esta topología tiene la característica de que existe una comunicación directa entre nodos, los cuales funcionan también como routers por lo que transmiten y reciben datos de otros nodos. Este tipo de estructura le permite ser altamente tolerante ante fallos ya que la información siempre pasara por todo el sistema y una falla de conexión en algún punto no generara perdida de data. El principal problema de este tipo de red es que dependiendo del número de nodos sensores la velocidad de transmisión se puede ver afectada y en general cada dispositivo consume una mayor cantidad energía que los presentados en la topología estrella. En la figura 2.4 se muestra un ejemplo de topología tipo malla.

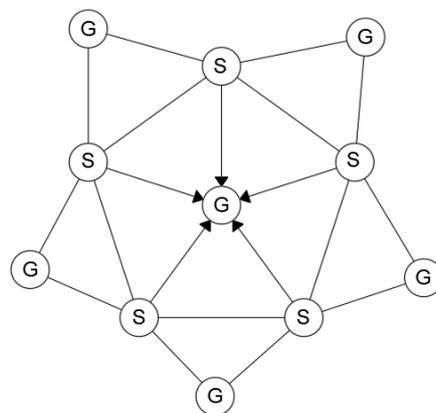


Figura 2.4 Topología malla

2.2.3. Arquitecturas IoT:

En la figura 2.5 se muestra el desarrollo de una arquitectura IoT la cual está compuesta de 4 etapas [15]:

- **Stage 1 Sensors/Actuadores:** Los sensores son los encargados de recolectar la información del ambiente y convertirlo en data para ser analizada, es el punto de inicio de una arquitectura IoT. Los actuadores pueden intervenir en la alteración de las condiciones físicas de la data analizada.
- **Stage 2 The Internet Gateway:** Es la etapa encargada de la recolección, procesamiento de los datos y su envío a través del internet al Stage 3.
- **Stage 3 Edge IT:** Una vez que los datos de IoT se han digitalizado y agregado, están listos para cruzar al ámbito de TI. Sin embargo, los datos pueden requerir un procesamiento adicional; aquí es donde entran en juego los sistemas informáticos de vanguardia, que realizan más análisis.
- **Stage 4 Data Center/Cloud:** Los datos que requieren un procesamiento más profundo se reenvían al centro de datos físico o sistemas basados en la nube, donde los sistemas de TI más potentes pueden almacenar, analizar y distribuir de forma segura la información.

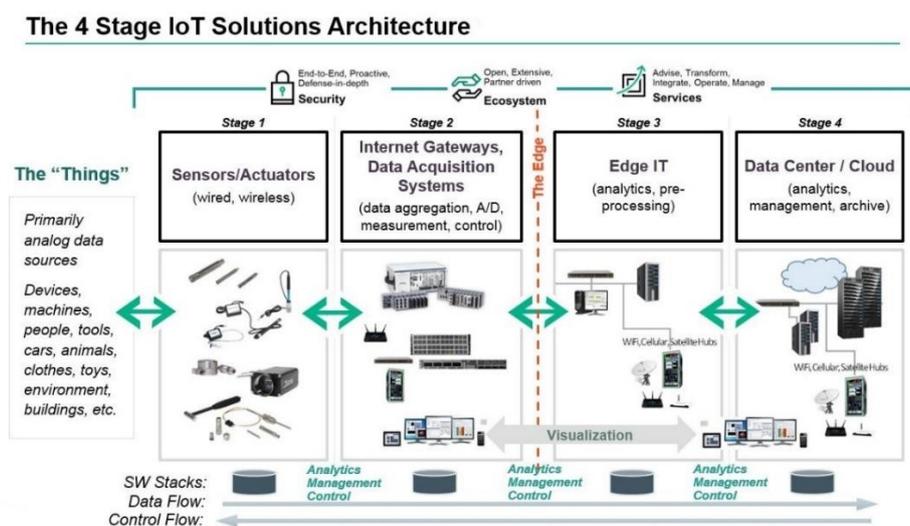


Figura 2.5 The 4 Stage IoT Solution Architecture

Fuente: [15]

2.3 Diseño de la arquitectura del software

2.3.1. Attribute-Driven Design

El método ADD es un proceso para definir la arquitectura de software donde el diseño se basa en los requerimientos de atributos de calidad del sistema. ADD sigue un proceso de diseño recursivo que divide un sistema o elementos del sistema aplicando patrones y tácticas arquitectónicas que satisfacen sus requisitos de conducción [16]. Como se aprecia en la figura 2.6, todo este proceso se realiza basándose en un ciclo de 7 pasos.

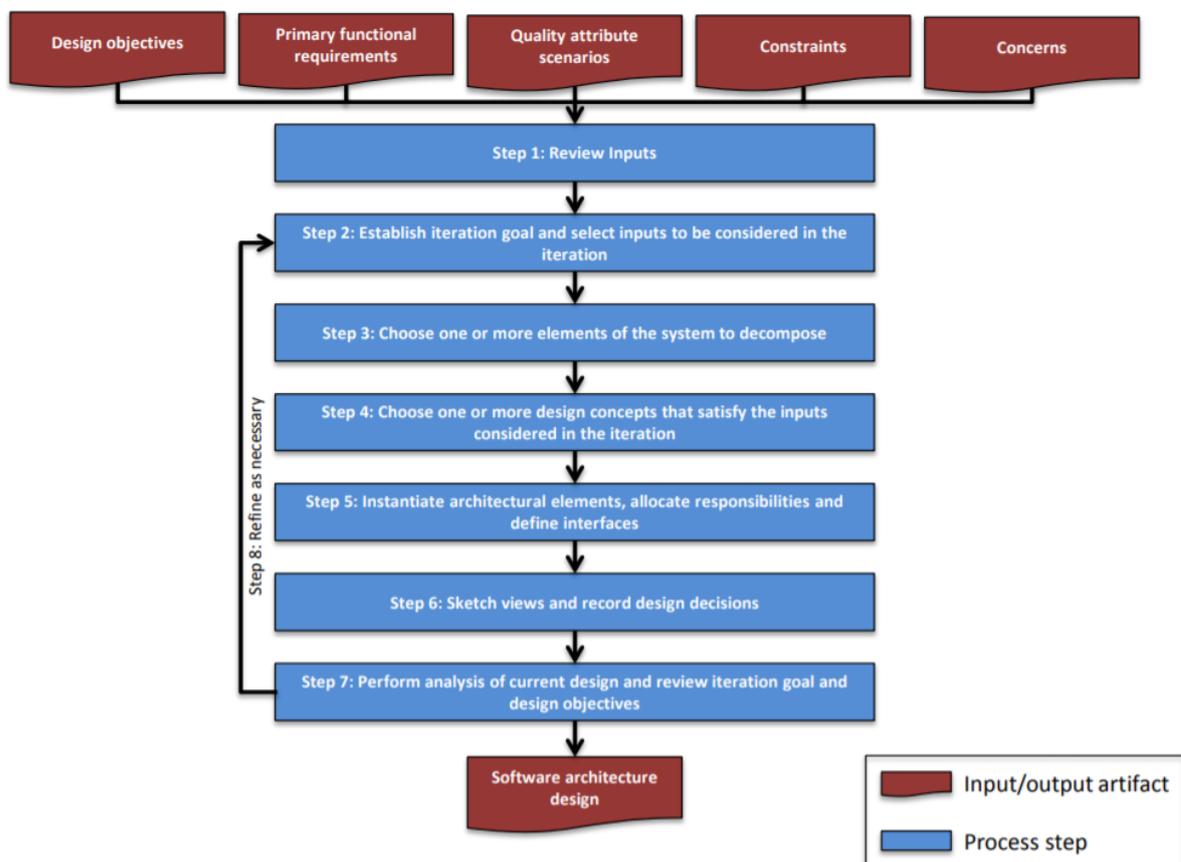


Figura 2.6 Proceso ADD 3.0

Fuente: [17]

Antes de empezar con los pasos del diseño según ADD se deben definir bien las entradas las cuales están divididas en propósito de diseño, requerimientos funcionales, atributos de calidad, restricciones y preocupaciones arquitecturales.

2.4. Productos comerciales:

2.4.1 IN_NOISE:

El IN_NOISE (figura 2.7) es un dispositivo inalámbrico para la monitorización y gestión de la contaminación por ruido. Fabricado siguiendo las especificaciones del estándar IEC 61672 para sonómetros [18].



Figura 2.7 Dispositivo IN_NOISE

Fuente: [18]

Principales características según el fabricante:

- Parámetro: LeqA.
- Sensibilidad del micrófono de 12,7 mV/Pa.
- Rango del sensor de sonido de 50 dBA a 100 dBA.
- Precisión de $\pm 0,5$ dBA (1kHz).
- Rango de frecuencias entre 20 Hz y 20 kHz.
- Micrófono omnidireccional.
- No cuenta con ponderación acústica.
- Modos de integración temporal rápida (125 ms) y lenta (1 s) configurable
- Tecnologías de comunicación: GPRS, SIGFOX, LoRa y ZigBee.

2.4.2. KUNAK NOISE N10:

El Kunak Noise N10 (figura 2.8) es un Sonómetro de tipo 2 que captura datos en tiempo real y los envía de forma inalámbrica a Kunak Cloud, el cual es un Software de Gestión del Ruido que analiza y configura de forma remota de los sensores [19].



Figura 2.8 Dispositivo Noise N10

Fuente: [19]

Principales características según el fabricante:

- Muestreo y envío de datos en tiempo real
- Altas correlaciones con Sonómetros de tipo 1
- Protección exterior IP65 contra el polvo y el agua
- Pequeño, ligero y fácil de instalar (< 20cm, <2kg)
- Plataforma Kunak Cloud para el análisis avanzado
- Alimentación por red eléctrica, panel solar y baterías
- Comunicación por GPRS, RS232

2.4.3. Sensor de ruido TA120:

El sensor de ruido TA120 (figura 2.9) es un dispositivo con precisión de sonómetro Clase 1 el cual cuenta con la protección de un kit de intemperie y la conectividad con la plataforma CESVA NoisePlatform y otras plataformas de código abierto [20].



Figura 2.9 Dispositivo TA120

Fuente: [20]

Principales características según el fabricante:

- Medición continua con precisión clase 1 según IEC 61672-1
- Protección con kit de exterior contra el viento, lluvia, pájaros e insectos
- Protección IP65 contra el polvo y el agua
- Dimensiones reducidas y fácil de integrar en el mobiliario urbano
- Alimentación por red eléctrica, POE (Power over Ethernet), 12 VDC (Paneles solares, baterías externas)
- Comunicación por Ethernet (RJ45), bucle 4-20mA, Wi-Fi.

2.5. Tesis y documentos:

2.5.1. The Implementation of Low-cost Urban Acoustic Monitoring Devices

El artículo técnico mencionado [21] presenta el desarrollo de un sensor de monitoreo acústico de bajo costo para la ciudad de Nueva York en Estados Unidos, el cual utiliza como núcleo de procesamientos un Tronsmart MK908ii como se aprecia en la figura 2.10.

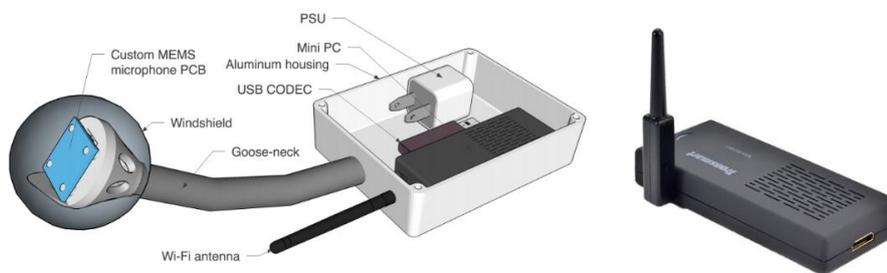


Figura 2.10 Prototipo de sensor con dispositivo Tronsmart MK908ii

Fuente: [21]

2.5.2. Desarrollo de un sistema de medición de ruido acústico

La tesis mencionada [22] presenta el desarrollo de un sistema con la capacidad de realizar mediciones ruido acústico y otras variables ambientales, para luego enviarlas hacia un sistema de administración de datos con interfaz de usuario. En la figura 2.11 se muestran los componentes físicos del sistema en los cuales destaca el uso de una

Raspberry Pi Model 2 para el procesamiento y transmisión de datos; y un sonómetro profesional clase 2 para la toma de datos sonoros.

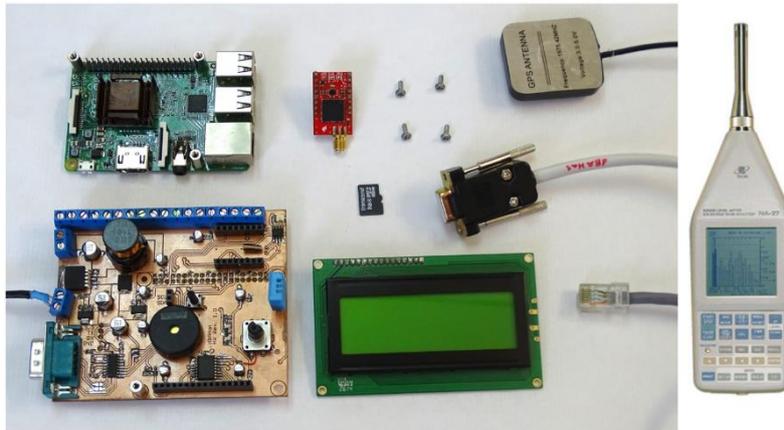


Figura 2.11 Componentes del sistema Hardware
Fuente: [22]

2.5.3. Diseño e implementación de una red inalámbrica de sensores inteligentes para el registro de la contaminación acústica

La Tesis mencionada [23] presenta el diseño e implementación de una red inalámbrica de sensores para el registro de la contaminación acústica en la Escuela Superior Politécnica de Chimborazo, utiliza una arquitectura similar a IoT como se aprecia en la figura 2.12 pero conectada a servidores locales. Para el procesamiento y transmisión de datos se utilizaron tarjetas Arduino Uno y Arduino Mega; el sensor de sonido utilizado fue un Sparkfun.

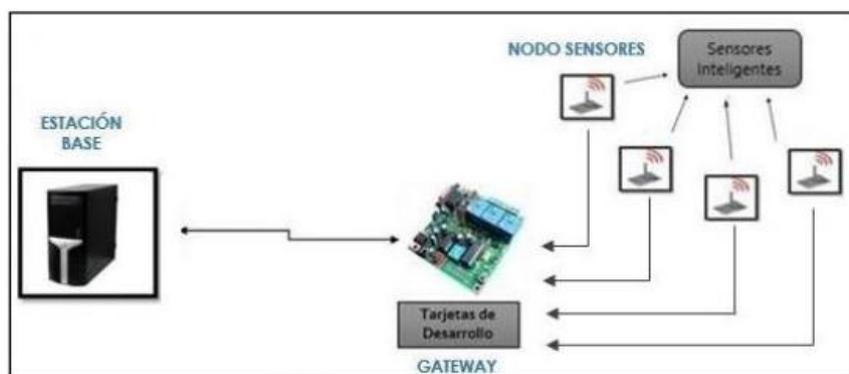


Figura 2.12 Red inalámbrica de sensores
Fuente: [23]

3. CAPÍTULO III: DISEÑO CONCEPTUAL DEL SISTEMA

En este capítulo se definirá el diseño conceptual del sistema ciberfísico, para esto se establecerán listas de requerimientos y se aplicarán las técnicas de diseño correspondientes a cada sector.

3.1. Selección de red IoT para sistema ciber

Para iniciar con el diseño conceptual del sistema se debe definir qué tipo de red IoT se va a utilizar ya que cada una tiene características particulares que acotarán el diseño de la solución final tanto en la parte física como en la ciber.

El sistema que se diseñará tiene las siguientes características:

- Los sensores deben ser capaces de operar en zonas donde no haya una red Wifi de internet.
- El rango de cobertura de la red debe ser capaz de transmitir entre edificaciones ya que se planea colocar los dispositivos en distintos puntos de la ciudad de Lima los cuales deben enviar la data a una misma base de datos.
- El consumo energético debe ser suficientemente bajo para poder mantener su continuidad, aunque se produzcan cortes energéticos a través de una batería de respaldo.
- Los dispositivos deben ser de bajo costo ya que este sistema servirá solo de referencia a los departamentos estatales encargados del control de la contaminación acústica para que puedan realizar estudios especializados de forma más eficiente.

Usando como base el análisis comparativo de las 3 principales redes IoT en la sección 2.2.1 de este documento se ha optado por una red LPWAN específicamente la tecnología LoRaWAN, ya que entre esta y SigFox, solo LoRaWAN posee una cobertura que le permite ser utilizada en el Perú. Adicionalmente un análisis realizado sobre redes de sensores LoRaWan demostró que este tipo de red logra cobertura hasta en zonas donde normalmente no llegan las redes 4G como lo son túneles o zonas donde las estructuras dificultan la llegada de estas redes [24].

3.1.1. Topología LoRaWAN:

LoRaWAN implementa una variación de la topología estrella mencionada en la sección 2.2.2 de este documento, esta es la topología estrella-estrella la cual se utilizan varios Gateways para retransmitir los datos a un servidor principal. En esta red, los nodos no están asociados a un solo puerto de enlace, sino que los datos enviados por un nodo son recibidos por distintos Gateways. Cada uno de estos Gateways reenviará la data desde el nodo final hasta el servidor de red en la nube.

La sección más compleja del proceso se ubica en el servidor de la red, ya que este se encarga de filtrar los paquetes redundantes recibidos, administrar la red, y realizar funciones de tasas de datos adaptables y seguridad [25]. En la figura 3.1 se muestra una gráfica con la topología descrita.

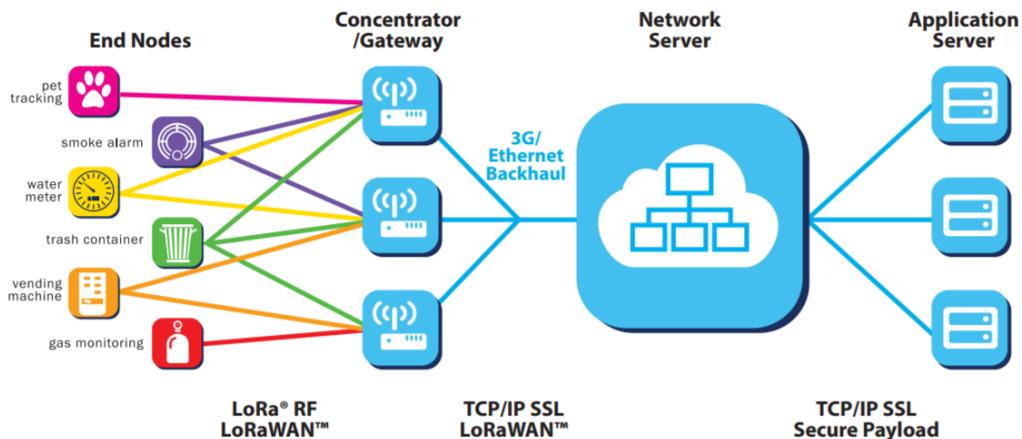


Figura 3.1 Topología LoRaWAN
Fuente: [25]

3.2. Diseño conceptual del sistema físico

3.2.1 Lista de requerimientos

El sistema físico para LoRaWAN está compuesto de un **Nodo Sensor** y un **Gateway** que podrán replicarse dependiendo de los requerimientos del cliente.

Tabla 2.1 Lista de Requerimientos Sistema Físico

Lista de Requerimientos		Elaborado por: Renzo Murata
Deseo Exigencia	Descripción	Responsable
Función principal		
Exigencia	Nodo sensor: <ul style="list-style-type: none"> Sensar los niveles de contaminación acústica de una zona urbana en tiempo real con un error no superior a ± 5dB y enviar la información al nodo Gateway. 	Renzo Murata
Exigencia	Gateway <ul style="list-style-type: none"> Recibir la información recolectada de varios nodos sensores y subirlos a la nube para su procesamiento. 	Renzo Murata
Geometría		
Exigencia	<ul style="list-style-type: none"> Dimensiones máximas de 200 mm x 200 mm x 100 mm tanto para el nodo sensor como para el Gateway. 	Renzo Murata
Cinemática		
Exigencias	<ul style="list-style-type: none"> El nodo sensor y el Gateway son estáticos. 	Renzo Murata
Fuerzas		
Exigencias	<ul style="list-style-type: none"> Los nodos sensores deben pesar no más de 3 kg Los Gateway deben pesar no más de 2 kg 	Renzo Murata
Energía		
Exigencias	Nodo Sensor: <ul style="list-style-type: none"> Fuente de alimentación primaria 220 VAC. Fuente de alimentación secundaria por batería recargable. Conversión de energía a voltajes no mayores no mayores a 5VDC para sensores y microcontroladores. 	Renzo Murata
Exigencias	Gateway: <ul style="list-style-type: none"> Fuente de alimentación primaria 220VAC Fuente de alimentación secundaria por batería recargable. Conversión de energía a voltajes no mayores no mayores a 5VDC para microprocesador. 	Renzo Murata
Señales		
Exigencias	Nodo Sensor: <ul style="list-style-type: none"> De entrada: <ul style="list-style-type: none"> Nivel de corriente analógica Presencia de lluvia Presión sonora analógica 	Renzo Murata

	<ul style="list-style-type: none"> • De Salida: <ul style="list-style-type: none"> ○ Estado de sensores ○ Presión sonora digital ponderada 	
Exigencias	Gateway: <ul style="list-style-type: none"> • De entrada: <ul style="list-style-type: none"> ○ Presión sonora digital ponderada ○ Estado de sensores • De Salida: <ul style="list-style-type: none"> ○ Estado de sensores ○ Presión sonora por Nodo Sensor 	Renzo Murata
Electrónica		
Exigencias	Nodo sensor: <ul style="list-style-type: none"> • Uso de sensores de sonido de bajo consumo compatibles con microcontroladores y con rango de frecuencia de 20Hz a 20 kHz. • Uso de módulos buzzer para revisar el estado de los sensores 	Renzo Murata
Comunicaciones		
Exigencias	<ul style="list-style-type: none"> • Conexión inalámbrica por radiofrecuencia entre nodos sensores y Gateway • Conexión por WIFI o Ethernet del nodo Gateway a la nube 	Renzo Murata
Seguridad		
Exigencias	<ul style="list-style-type: none"> • Nodo sensor y Gateway protegido contra la humedad y el polvo con grado de protección IP65 o más. 	Renzo Murata
Ergonomía		
Exigencias	<ul style="list-style-type: none"> • Diseño compacto de fácil transportabilidad. • Fácil acceso a componentes electrónicos para montaje y reemplazo 	Renzo Murata
Fabricación		
Exigencias	<ul style="list-style-type: none"> • Que se pueda fabricar o maquinar en talleres locales. • Fabricable con los equipos y herramientas del mercado nacional. 	Renzo Murata
Control de calidad		
Exigencias	<ul style="list-style-type: none"> • Sistema de verificación de funcionamiento y calibración de sensores 	Renzo Murata
Montaje		
Exigencias	<ul style="list-style-type: none"> • El montaje de sensores no requerirá más de 2 personas • Los nodos sensores deberán tener una compuerta de fácil acceso para su mantenimiento. 	Renzo Murata

Transporte		
Exigencias	<ul style="list-style-type: none"> • Sensores de fácil transporte que podrán ser llevados en cajas pequeñas. • Debe poder ser llevado por 1 persona. 	Renzo Murata
Uso		
Exigencias	<ul style="list-style-type: none"> • En un rango de 0°C a 55°C debe garantizar un rendimiento óptimo • Para un uso permanente con intervalos de mantenimiento semestrales. • Para uso urbano en calles concurridas de Lima 	Renzo Murata
Mantenimiento		
Exigencias	<ul style="list-style-type: none"> • Limpieza semanal. • Mantenimiento semestral 	Renzo Murata
Costos		
Exigencias	Costo de fabricación: <ul style="list-style-type: none"> • No más de \$250 por Gateway • No más de \$200 por Nodo Sensor 	Renzo Murata

3.2.2. Estructura de funciones

EL sistema físico está compuesto por el nodo sensor y el Gateway por lo que cada uno tendrá su propia estructura de funciones.

Energía:

- **Nodo Sensor:** El flujo de energía se obtiene mediante 2 fuentes: una fuente principal AC/DC de 5V y una batería recargable. Este sistema garantizará el funcionamiento del sistema, aunque se produzcan cortes en el servicio eléctrico, ambas fuentes pasan al bloque de potencia.
- **Gateway:** El flujo de energía se obtiene mediante 2 fuentes: una fuente principal AC/DC de 5V y una batería recargable. Este sistema garantizará el funcionamiento del sistema, aunque se produzcan cortes en el servicio eléctrico, ambas fuentes pasan al bloque de potencia.

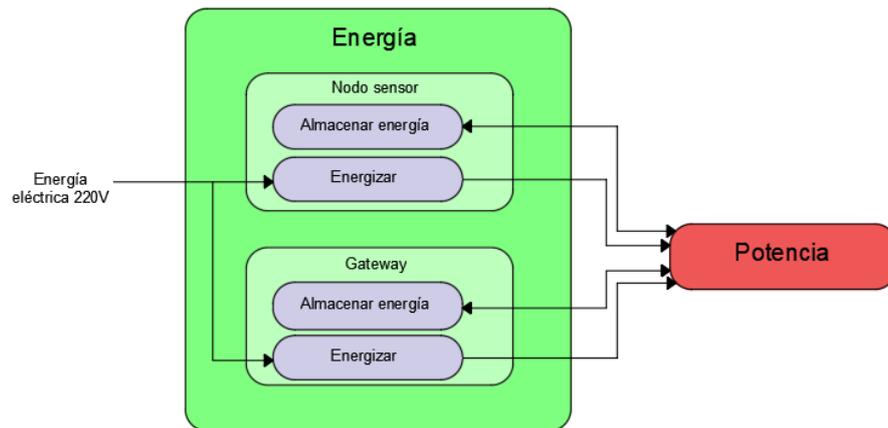


Figura 3.2 Bloque Energía

Potencia:

- **Nodo Sensor:** En el bloque de potencia las 2 fuentes de energía pasan a un selector que en primera instancia permitirá el paso de corriente de la fuente principal y en caso de pérdida de energía pasará a usar la fuente secundaria.
- **Gateway:** En el bloque de potencia las 2 fuentes de energía pasan a un selector que en primera instancia permitirá el paso de corriente de la fuente principal y en caso de pérdida de energía pasará a usar la fuente secundaria.

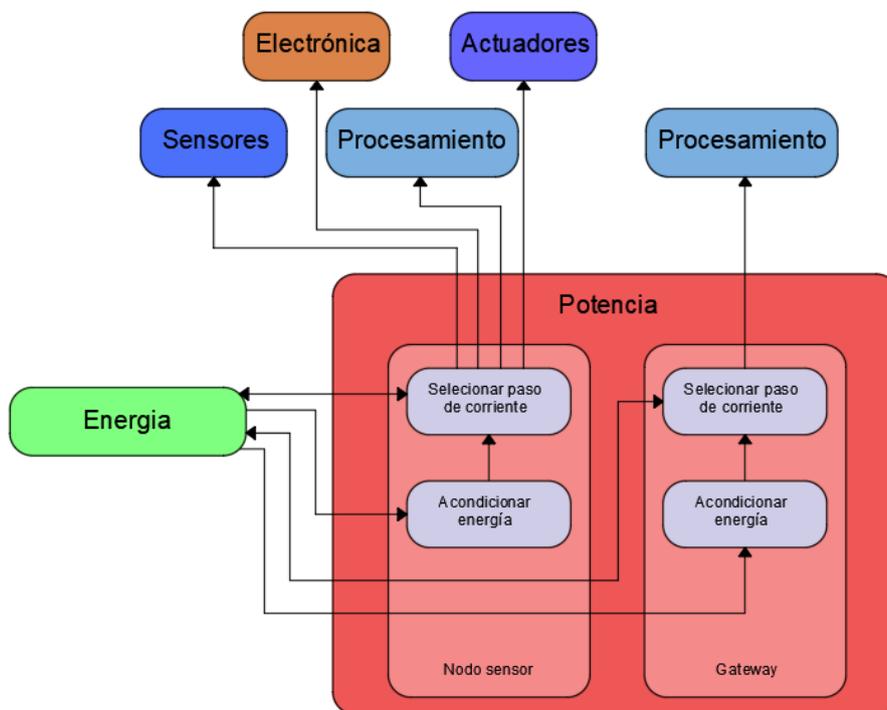


Figura 3.3 Bloque Potencia

Sensores:

- **Nodo Sensor:** En este bloque se encuentra el sensor de sonido que se encargara de medir el nivel de presión sonora y transmitirlo a través de una salida analógica, un sensor de lluvia que indicara la presencia o ausencia de esta a través de una señal digital, y un sensor de corriente para saber si la fuente primaria provee energía.

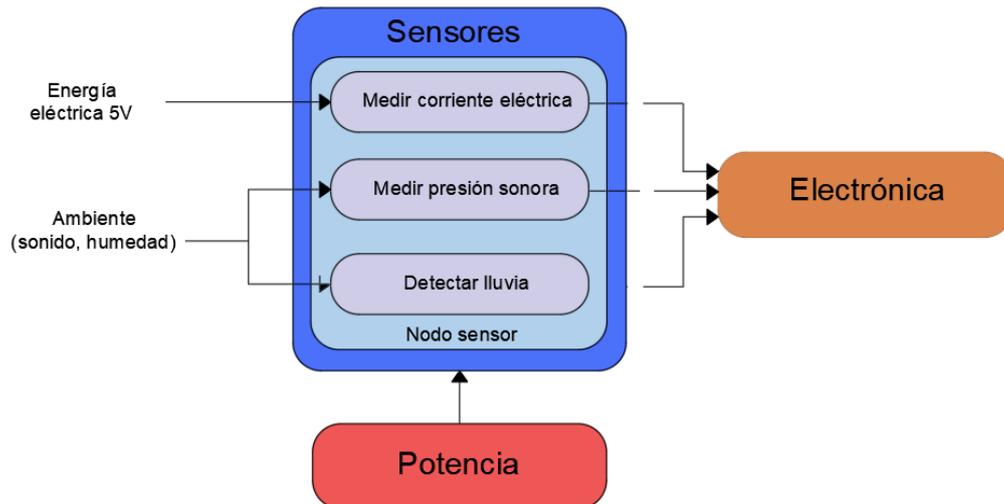


Figura 3.4 Bloque Sensores

Electrónica:

- **Nodo Sensor:** Este bloque se encargará de acondicionar las señales provenientes de los sensores lo cual implica amplificación en caso del sensor de sonido, eliminación de ruidos y escalamiento de voltaje o corriente para que sea interpretado correctamente por el sistema de control.

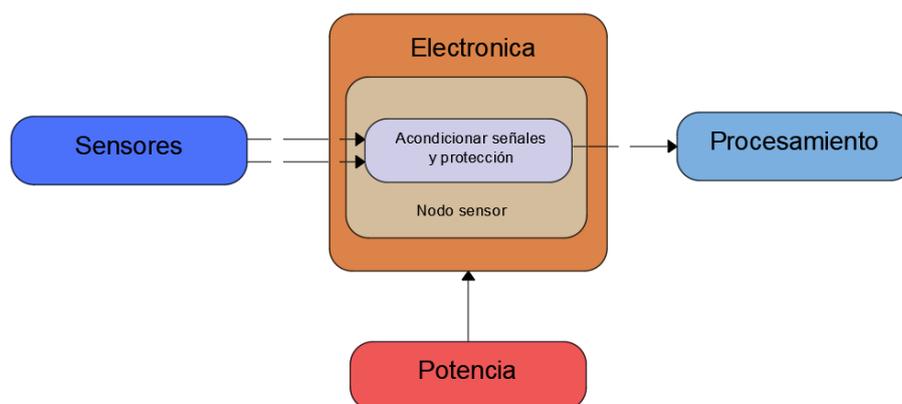


Figura 3.5 Bloque Electrónica

Procesamiento:

- **Nodo Sensor:** Este bloque se encargará de interpretar las señales analógicas y digitales provenientes de la electrónica, procesarla, validarla, transmitirla y accionar actuadores en caso sea necesario.
- **Gateway:** Este bloque se encargará de recolectar la data de los nodos sensores, validarla y prepararla para su transmisión al sistema Ciber.

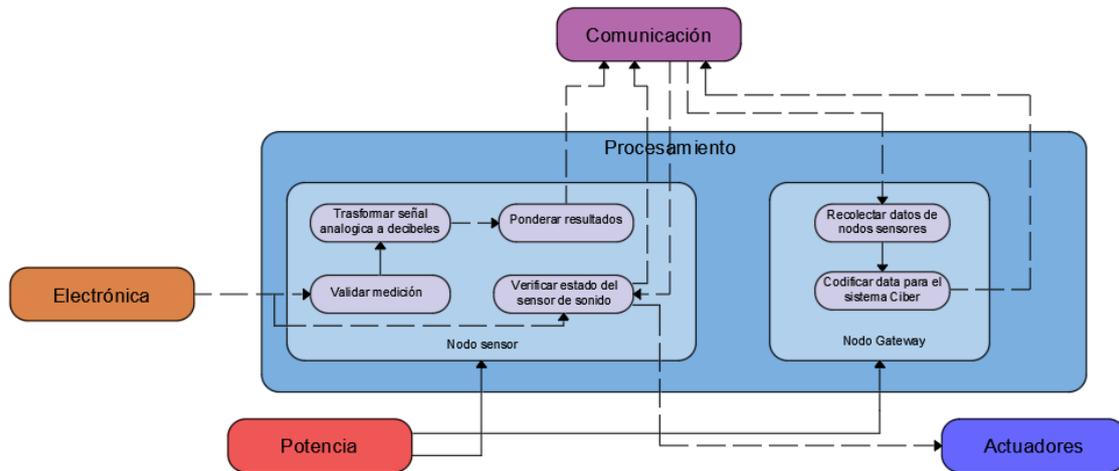


Figura 3.6 Bloque Control

Actuadores:

- **Nodo Sensor:** En este bloque se dispondrá de un buzzer para verificar el estado del sensor de sonido.

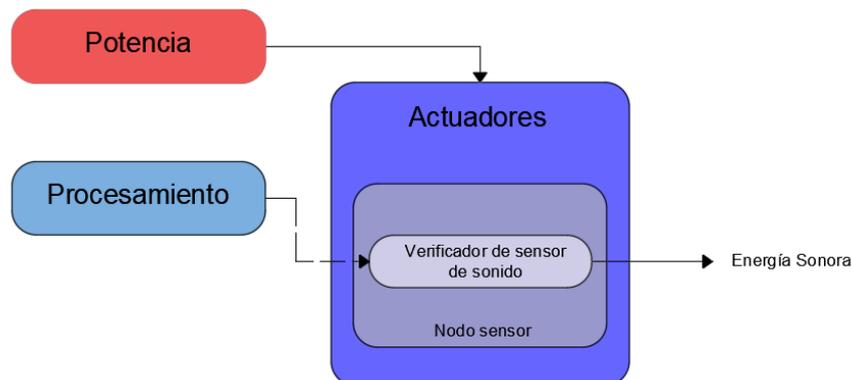


Figura 3.7 Bloque Comunicación

Comunicación:

- **Nodo Sensor:** En este bloque el nodo sensor cuenta con un transceptor de radio frecuencia que se encarga de transmitir la información procesada al Gateway y recibir los pedidos de revisión de sensores por parte del sistema ciber.

- Gateway: En este bloque el Gateway cuenta con dos modos de comunicación: un transceptor de radiofrecuencia para enviar y recibir información de los nodos sensores y una conexión a internet para enviar la información al sistema ciber.

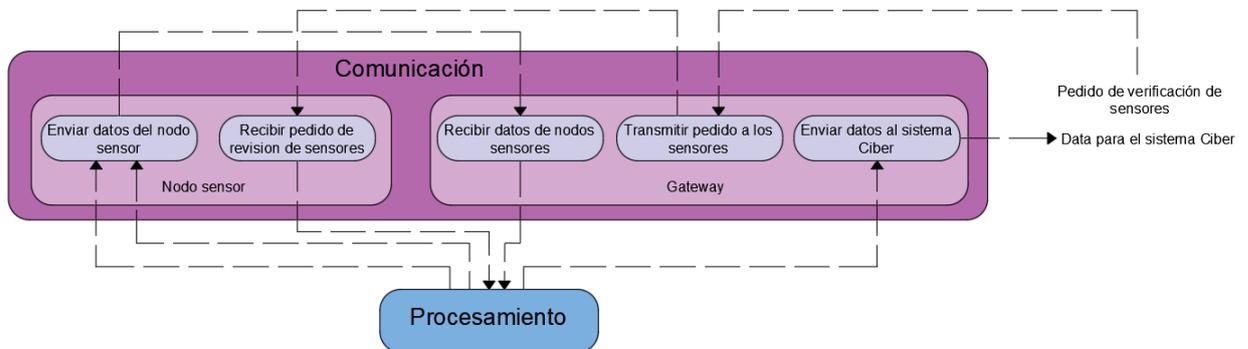


Figura 3.8 Bloque Comunicación

Mecánica:

- Nodo sensor y Gateway: Este bloque se muestra las estructuras mecánicas que permitirán el correcto funcionamiento de sistema ante la intemperie para garantizar su durabilidad y efectividad.

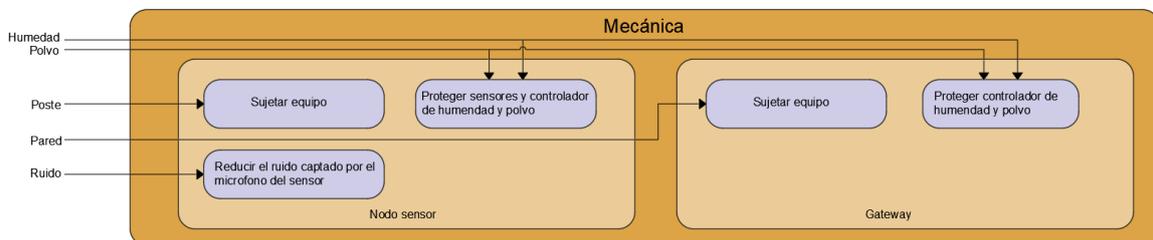


Figura 3.9 Bloque Mecánica

Para vista completa de la estructura de funciones revisar anexo 1.

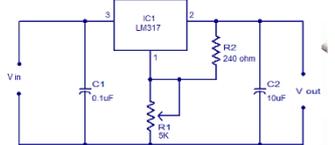
3.2.3. Matriz morfológica

A partir de la estructura de funciones del sistema físico se realizarán dos matrices morfológicas (Nodo sensor y Gateway) donde se ilustrarán los componentes físicos y lógicos que cumplen con cada función.

Opción 1	Opción 2	Opción 3
→	→	→

Nodo sensor:

Tabla 3.1 Matriz morfológica Nodo Sensor

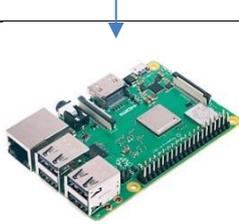
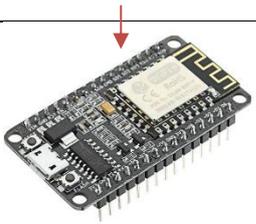
Función		Alternativa 1	Alternativa 2	Alternativa 3	
Energía	Energizar	 Fuente conmutada 5V	 +  Fuente conmutada 12V + circuito regulador VDC		
	Almacenar energía	 Lithium ion battery	 Bateria de níquel cadmio	 Lipo battery	
Potencia	Seleccionar paso de corriente	 Placa UPS	 Modulo relevaror		
	Acondicionar energía				
Sensores y electrónica	Medir corriente eléctrica		 Sesor de corriente		
	Medir presión sonora	 Detector de sonido	 Detector de decibeles	 Sonometro integrador	
	Detectar lluvia	 Sensor de lluvia por variacion de conductividad de bajo consumo		 Sensor de lluvia por variacion de conductividad industrial	
Procesamiento	Hardware	Validar medición	 Arduino Nano	 STM32 + DS3231	 Tiva Launchpad
		Transformar señal analógica a decibeles			
		Ponderar resultados			
		Verificar estado del sensor de sonido			

Software	Validar medición	Revisión de señales enviadas por los sensores
	Transformar señal analógica a decibeles	Escalar valores de entrada analógicos a decibeles
	Ponderar resultados	Aplicar fórmula de ponderación
	Verificar estado del sensor de sonido	Revisión de datos recibidos por el transceptor, señales enviadas por los sensores y accionamiento de actuadores
Actuadores	Verificador de sensor de sonido	 <p>Buzzer</p>
Comunicación	Enviar datos del nodo sensor	 <p>Lora RF</p>
	Recibir pedido de revisión de sensores	
Mecánica	Sujetar equipo	 <p>Abrazadera de sección circular</p>
	Proteger sensores y controlador contra humedad y polvo	 <p>Gabinete Rittal</p>
	Reducir ruido captado por micrófono	 <p>Filtro cortaviento protector para sonómetro</p>

Gateway:

Tabla 3.2 Matriz morfológica Gateway

		Función	Alternativa 1	Alternativa 2	Alternativa 3
Energía	Energizar		 Fuente conmutada 5V	 Fuente lineal	 Fuente conmutada 5V
	Almacenar energía		 Lithium ion battery	 Bateria de níquel cadmio	 Lipo battery
Potencia	Seleccionar paso de corriente		 Placa UPS	 Placa para batería	 Modulo releador
	Acondicionar energía				
Procesamiento	Hardware	Recolectar datos de nodos sensores	 Raspberry Pi	 ESP8266	 Gateway comercial
	Software	Codificar data para el sistema ciber	Revisión de data recibida por el transceptor	Código de programación proporcionado por servidor TTN	Programación según fabricante
Comunicación	Hardware	Recolectar datos de nodos sensores	 Lora RF Multiples canales	 Lora RF 1 canal	 Gateway comercial
	Software	Transmitir pedido a los sensores			

	Enviar data al sistema Ciber	 Raspberry Pi	 ESP8266	
Mecánica	Sujetar equipo	 Soporte para pared		
	Proteger Gateway contra humedad y polvo	 Gabinete Rittal		

3.2.4. Conceptos de solución

Solución 1:

La solución 1 utiliza, tanto para el nodo sensor como para el Gateway, una fuente de alimentación conmutada para transformar la energía de 220VAC a 5VDC que requieren los elementos electrónicos. Esta fuente ira conectada a una placa UPS que dispone de un circuito integrado que alimentara al microcontrolador con la energía suministrada por la fuente y en caso de pérdida de tensión pasara a energizar con la batería de litio acoplada a él.

Para la detección del nivel de presión sonora se utilizará un sensor de sonido el cual a través de un pin analógico informa al controlador sobre la amplitud de la onda sonora sensada. Según la resolución ministerial N°227-2013-MINAM, el sensado de contaminación acústica debe hacerse en momentos donde no haya presencia de lluvia por lo que el nodo sensor dispondrá de un sensor de la misma para identificar cuando no se podrán registrar datos, en esta solución se utilizará un sensor de lluvia por variación de conductividad de bajo consumo. El microcontrolador seleccionado fue el arduino nano el cual posee pines analógicos capaces de recibir la información del sensor de sonido y un reloj en tiempo real que permitirá programar una revisión diaria del estado de los sensores.

Este estado podrá ser revisado a través de un buzzer ubicado también en el nodo, este dispositivo será accionado por el controlador y emitirá un sonido el que sensor deberá identificar y enviar al Gateway.

La transmisión entre el Nodo Sensor y el Gateway se dará por radiofrecuencia a través de módulos Lora. Ya que el Gateway debe ser capaz de recibir múltiples señales en simultaneo, se utilizará un módulo RAK2245 Pi Hat el cual permite configura el Raspberry como un Gateway Lora con recepción de hasta 8 canales en simultaneo. Para transmitir la data recolectada del Gateway al sistema Ciber se utilizará la antena Wifi integrada en la raspberry y una red de internet presente en su zona.

En el ámbito mecánico la sujeción del Nodo sensor será por una abrazadera circular la cual permitirá situarlos en postes de alumbrado o semáforos. Para su protección contra la intemperie se utilizará un gabinete Rittal con calificación IP65 de protección de equipos electrónicos, este gabinete será adaptado para poder colocar una antena externa, una salida para el micrófono y el sensor de lluvia. El micrófono también dispondrá de una esponja protectora capas de filtrar el ruido generado por el viento y proteger al micrófono del polvo y la humedad. En el caso del nodo Gateway la protección será igualmente por un gabinete Rittal con protección IP65.

En la figura 3.10 se muestra el bosquejo de la solución descrita:

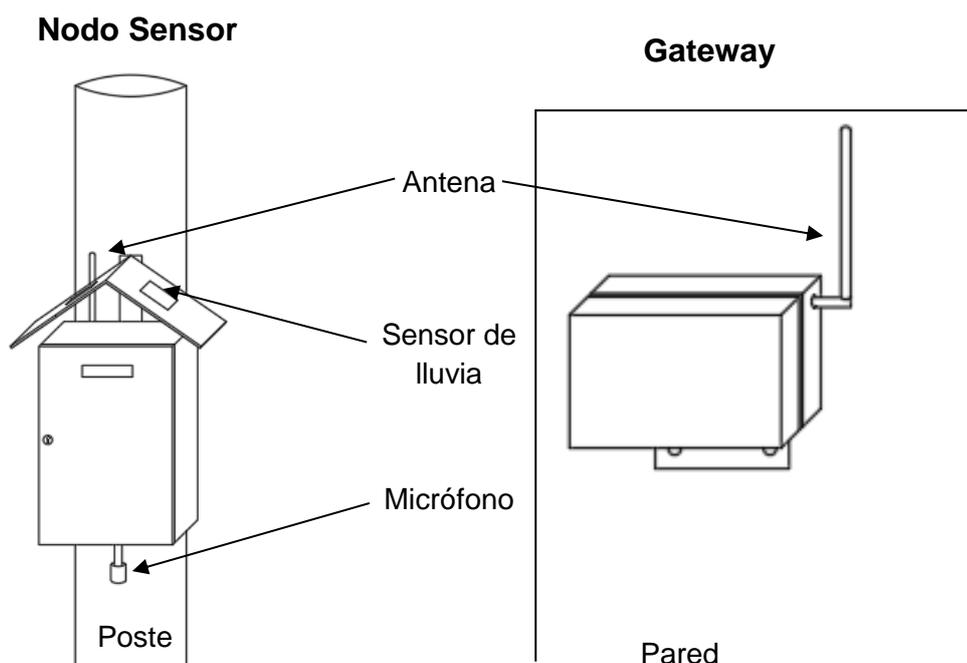


Figura 3.10 Diseño Conceptual Solución 1

Solución 2:

La solución 2 utiliza, en el nodo sensor, una fuente de alimentación conmutada para transformar la energía de 220VAC a 5VDC que piden los elementos electrónicos. Esta fuente y una batería de níquel-cadmio irán conectadas a través de un circuito con módulos relevadores que alimentarán al microcontrolador principalmente con la energía suministrada por la fuente y en caso de pérdida de tensión con la batería mencionada. Para el Gateway la alimentación será a través de un adaptador a 3.3VDC y una batería níquel cadmio, ambas irán conectadas a través de un circuito con módulos relevadores al igual que el nodo sensor.

Para la detección del nivel de presión sonora se utilizará un sensor con circuito integrado que entrega la data directamente en decibeles sin ponderación. Siguiendo la norma N°227-2013-MINAM al igual que en las soluciones anteriores, se utilizará un sensor de lluvia de bajo consumo que será alimentado por la fuente conmutada y avisará al microcontrolador la presencia de lluvia. El microcontrolador seleccionado fue el STM32, el cual posee pines analógicos y digitales capaces de recibir la información de los sensores. Adicionalmente se le acoplará un módulo RTC DS3231 para poder programar una revisión de estado de sensores, este podrá ser revisado a través de un módulo buzzer ubicado también en el nodo. El dispositivo será accionado por el controlador y emitirá un sonido el que sensor deberá identificar y enviar al Gateway.

La transmisión entre el Nodo Sensor y el Gateway se dará por radiofrecuencia a través de módulos Lora. Para esta solución se diseñará un Gateway económico de un solo canal el cual puede ser conectada a varios nodos sensores, pero no podrá recibir data en simultaneo. Se utilizará como controlador el ESP8266, el cual estará conectado a un módulo RFM95 para captar las señales de radio de los nodos sensores. La transmisión de la información al sistema Ciber se realizará por la antena wifi integrada del ESP8266.

En el ámbito mecánico la sujeción del Nodo sensor será por una abrazadera circular la cual permitirá situarlos en postes de alumbrado o semáforos. Para su protección contra la intemperie se utilizará un gabinete de Rittal con calificación IP65 de protección de equipos electrónicos, este gabinete será adaptado para poder colocar una antena externa, una salida para el micrófono y el sensor de lluvia. El micrófono también dispondrá de una esponja protectora capas de filtrar el ruido generado por el viento y proteger al micrófono

del polvo y la humedad. El Gateway también poseerá un gabinete Rittal con calificación IP65.

En la figura 3.11 se muestra el bosquejo de la solución descrita:

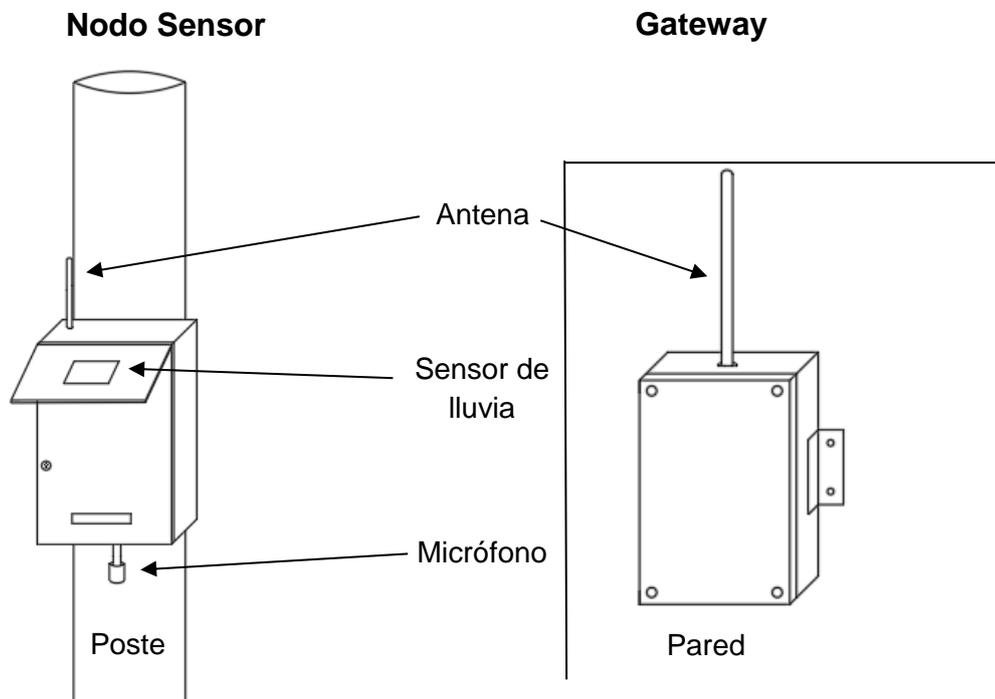


Figura 3.11 Diseño Conceptual Solución 2

Solución 3:

La solución 3 utiliza, en el nodo sensor, una fuente de alimentación conmutada para transformar la energía de 220VAC a 12VDC y esta pasará a un regulador de voltaje a 5 VDC que piden los elementos electrónicos. Esta fuente y una batería LiPo irán conectadas a través de un circuito con módulos relevadores que alimentarán al microcontrolador principalmente con la energía suministrada por la fuente y en caso de pérdida de tensión con la batería mencionada. El Gateway será alimentado por una fuente conmutada a 5 VDC con el mismo circuito selector y baterías LiPo.

Para la detección del nivel de presión sonora se utilizará un sonómetro integrador clase 2 al cual se le adaptará una comunicación con el microcontrolador para enviar la data sensada. Siguiendo la norma N°227-2013-MINAM al igual que en la solución 1, se utilizará un sensor de lluvia de tipo industrial que será alimentado por la fuente conmutada y enviará una señal digital al controlador en caso de lluvia (este dispositivo solo funcionará con la alimentación principal en operación). El microcontrolador seleccionado fue el Tiva Launchpad el cual posee pines analógicos y digitales capaces de recibir la

información de los sensores de sonido y lluvia, y un reloj en tiempo real que permitirá programar una revisión diaria del estado de los sensores. Este estado podrá ser revisado a través de un módulo buzzer ubicado también en el nodo. El dispositivo será accionado por el controlador y emitirá un sonido el que sensor deberá identificar y enviar al Gateway.

La transmisión entre el Nodo Sensor y el Gateway se dará por radiofrecuencia a través de módulos Lora. Ya que el Gateway debe ser capaz de recibir múltiples señales en simultáneo, se utilizará el Lorank 8, un Gateway comercial diseñado para comunicación LoraWan. Este Gateway también posee conexión Ethernet con la que se enviara la data recolectada al sistema Ciber.

En el ámbito mecánico la sujeción del Nodo sensor será por una abrazadera circular la cual permitirá situarlos en postes de alumbrado o semáforos. Para su protección contra la intemperie se utilizará un gabinete de plástico con calificación IP65 de protección de equipos electrónicos, este gabinete será adaptado para poder colocar una antena externa, una salida para el micrófono y el sensor de lluvia. El micrófono también dispondrá de una esponja protectora capas de filtrar el ruido generado por el viento y proteger al micrófono del polvo y la humedad. En el caso del nodo Gateway no será necesario un diseño porque el dispositivo viene preparado para trabajos a la intemperie.

En la figura 3.12 se muestra el bosquejo de la solución descrita:

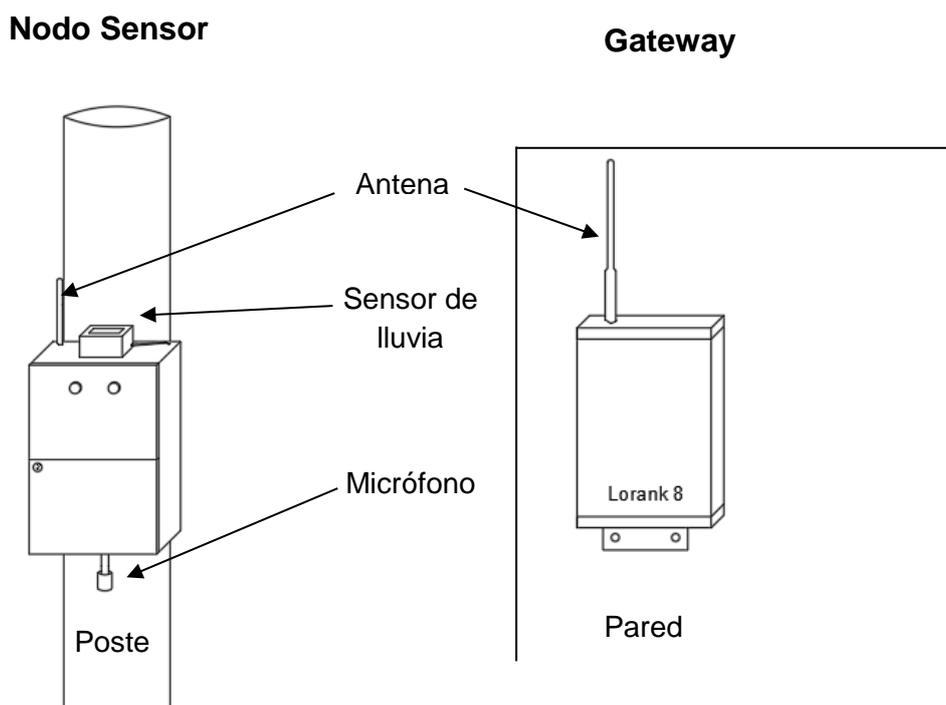


Figura 3.12 Diseño Conceptual Solución 3

3.2.5. Evaluación Técnico-Económica

De acuerdo a la norma VDI 2225, se establecen distintos criterios de evaluación a las soluciones propuestas, evaluando el valor técnico y el valor económico. En este sistema se asignará un puntaje a cada criterio y cada criterio tendrá un peso en función a su importancia. Finalmente sumando los valores obtenidos se obtendrá un puntaje final, el cual será ponderado en función a una solución ideal.

Tabla 3.3 Valor Técnico del sistema

Valor Técnico									
P: Puntaje de 0 a 4 (Escala de valores según VDI2225)									
0 = No satisface, 1 = Aceptable, 2 = Suficiente, 3 = Bien, 4 = Ideal									
G: Peso ponderado que se da en función a los criterios de evaluación									
Soluciones		Solución 1		Solución 2		Solución 3		Solución ideal	
Criterios de evaluación	G	P	GP	P	GP	P	GP	P	GP
Funcionalidad	2	3	6	3	6	3	6	4	8
Velocidad de operación	1	2	2	2	2	2	2	4	4
Seguridad	2	3	6	2	4	2	4	4	8
Precisión	1	3	3	2	2	3	3	4	4
Calidad	1	2	2	1	1	3	3	4	4
Montaje	1	3	3	3	3	3	3	4	4
Puntaje			22		18		21		32
Valor técnico ponderado X		0.688		0.563		0.656		1	
Orden		1		3		2			

Tabla 3.4 Valor económico del sistema

Valor Económico									
P: Puntaje de 0 a 4 (Escala de valores según VDI2225)									
0 = No satisface, 1 = Aceptable, 2 = Suficiente, 3 = Bien, 4 = Ideal									
G: Peso ponderado que se da en función a los criterios de evaluación									
Soluciones		Solución 1		Solución 2		Solución 3		Solución ideal	
Criterios de evaluación	G	P	GP	P	GP	P	GP	P	GP
Eficiencia energética	2	2	4	3	6	1	2	4	8
Costo de tecnología	2	2	4	3	6	1	2	4	8
Costo de estructura	1	3	3	3	3	3	3	4	4
Costo mantenimiento	1	3	3	3	3	2	2	4	4
Puntaje			14		18		9		24
Valor económico ponderado Y		0.583		0.75		0.375		1	
Orden		2		1		2			

Con la data recolectada se realiza una gráfica de evaluación Técnico-Económico como se aprecia en la figura 3.13. Tras analizar esta grafica se puede concluir que la solución más óptima sería la solución 1 ya que se acerca más a la línea ideal la cual representa el balance deseado entre el valor técnico y el económico.

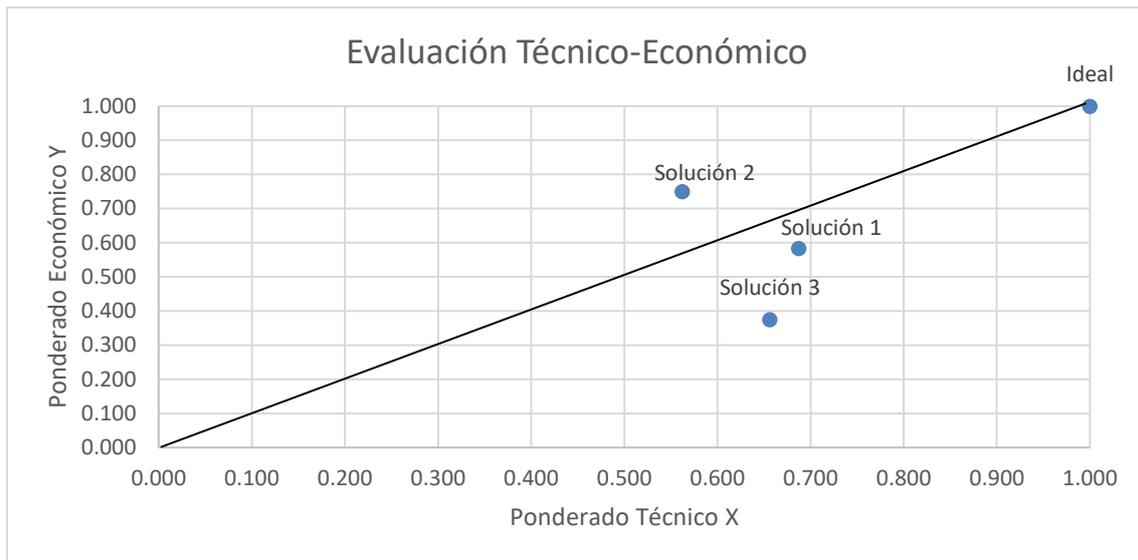


Figura 3.13 Evaluación técnico económica

3.3. Diseño conceptual del sistema ciber

3.3.1 Lista de requerimientos

Tabla 3.5 Lista de requerimientos Sistema Ciber

Lista de Requerimientos		Elaborado por: Renzo Murata
Deseo Exigencia	Descripción	Responsable
Funcional		
Exigencias	<ul style="list-style-type: none"> Procesar los datos de contaminación acústica y ofrecer un entorno visual para analizarlos Tener un control de la ubicación de los nodos sensores Registrado con fecha de todos los eventos captados por los sensores Informar al usuario en caso de niveles dañinos de presión sonora 	Renzo Murata
Comercial		
Exigencias	<ul style="list-style-type: none"> El costo de diseño de la arquitectura no debe exceder los S/. 3000 Los costos de los servicios en la nube no deben exceder los \$100 mensuales 	Renzo Murata
Humano		
Exigencias	<ul style="list-style-type: none"> Acceso público mediante una interfaz móvil de usuario Acceso privado mediante una Interfaz web de usuario. 	Renzo Murata
Fiabilidad		
Exigencias	<ul style="list-style-type: none"> Administración del acceso a los servicios y recursos de la nube a través de permisos y credenciales. Tecnología CHIRP (Compressed High Intensity Radar Pulse) para transmisión de datos. Base de datos replicada 	Renzo Murata
Tiempo de respuesta		
Exigencias	<ul style="list-style-type: none"> Intervalos de tiempo entre eventos de la arquitectura deben estar en el orden de milisegundos. Latencia tolerable < 5 segundo. 	Renzo Murata
Datos		
Exigencias	<ul style="list-style-type: none"> Identificación de datos por sensor a través del Network Session Key. 	Renzo Murata

4. CAPÍTULO IV: DISEÑO DEL SISTEMA CIBERFÍSICO INTEGRADO

4.1. Funcionamiento del sistema integrado

Ya teniendo definido una estructura física definida y los requerimientos de la parte ciber, se pasará a explicar el funcionamiento en conjunto de estos. El sistema busca mantener informado a los usuarios los niveles de contaminación acústica en tiempo real de las zonas donde se encuentren transitando, además se podrá tener un registro de todas las mediciones tomadas las cuales serán mostradas a través de gráficos que permitirán identificar zonas críticas, información que podrá ser utilizada por las organizaciones estatales respectivas para tomar medidas al respecto.

Para realizar este sensado, basándose en el protocolo nacional de monitoreo ambiental [26], los nodos sensores enviarán cada minuto el nivel ponderado de contaminación acústica de su respectiva zona. Esta data se obtendrá a través de la captura de 60 muestras (en intervalos de 1 segundos), las cuales generarán una aproximación a la medida ponderada tipo A a través de la siguiente fórmula:

$$L_{AeqT} = 10 \log \left[\frac{1}{n} \sum_{i=1}^n 10^{0.1L_i} \right].$$

Donde

- L_{AeqT} = Nivel de presión sonora ponderada A en un tiempo T.
- n = Cantidad de mediciones en el periodo T
- L_i = Nivel de presión sonora i capturada en el tiempo de muestreo.

El protocolo indica que no se deben realizar medidas cuando existen condiciones meteorológicas que puedan afectar la calidad de los datos. En el caso de la ciudad de Lima, esta condición vendría a ser la lluvia por lo que los nodos también cuentan con un sensor de lluvia que enviara un dato de presencia o ausencia de esta junto con la presión sonora para que pueda ser tomada en cuenta al momento de enviar la data a la nube. Adicionalmente el sistema cuenta con un buzzer con el que el usuario administrador podrá revisar el funcionamiento de los sensores de sonido de manera remota a través de ordenes enviadas desde el servidor Cloud.

Por otra parte, la arquitectura IoT del sistema ciber será diseñada en base a la metodología ADD (Attribute Driven Design) que tomará en cuenta los atributos de calidad, la funcionalidad primaria, las preocupaciones arquitecturales y las limitaciones del sistema. Este diseño permitirá recibir y procesar tramas de datos, almacenar información, mostrar información mediante gráficas y actualizar valores en tiempo real que podrán mostrar la contaminación sonora a través de páginas web y aplicativos móviles.

4.2. Diseño del sistema físico

En esta sección se realizará el diseño del nodo sensor y del Gateway, para lo cual se definirá su arquitectura, modelo 3D, se seleccionarán los componentes y se realizarán diagramas de flujo que muestren el funcionamiento de sus programas.

4.2.1 Arquitectura del hardware

En la figura 4.1 y 4.2 se muestra un diseño general de la arquitectura física del sistema físico tanto para el Nodo Sensor como para el Gateway

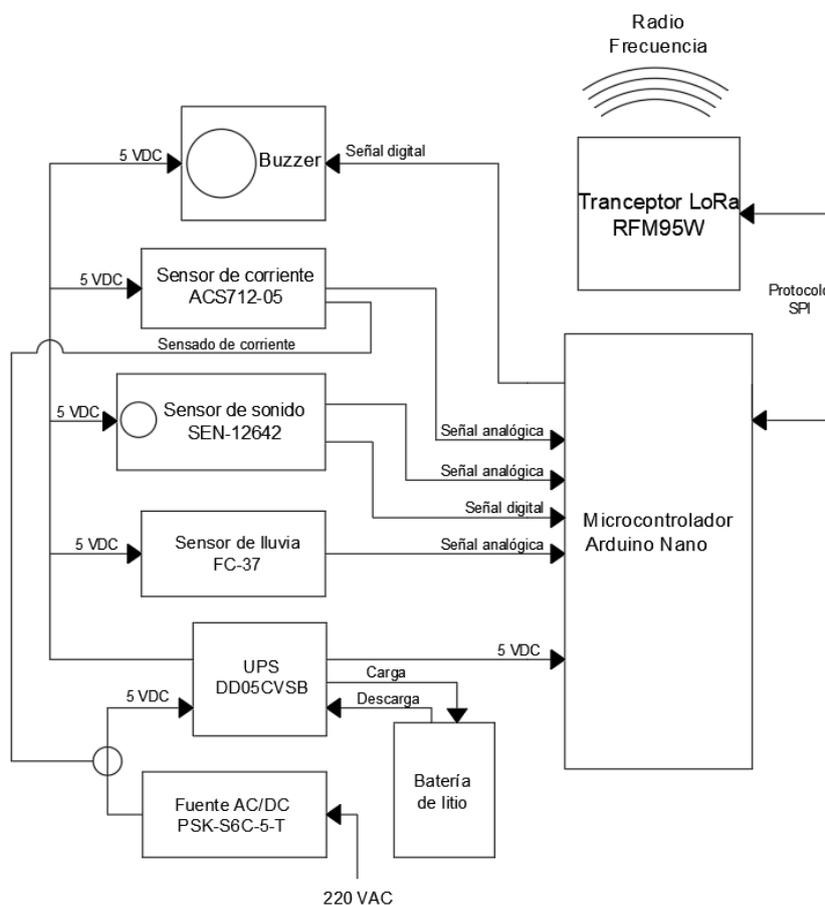


Figura 4.1 Arquitectura de hardware Nodo Sensor

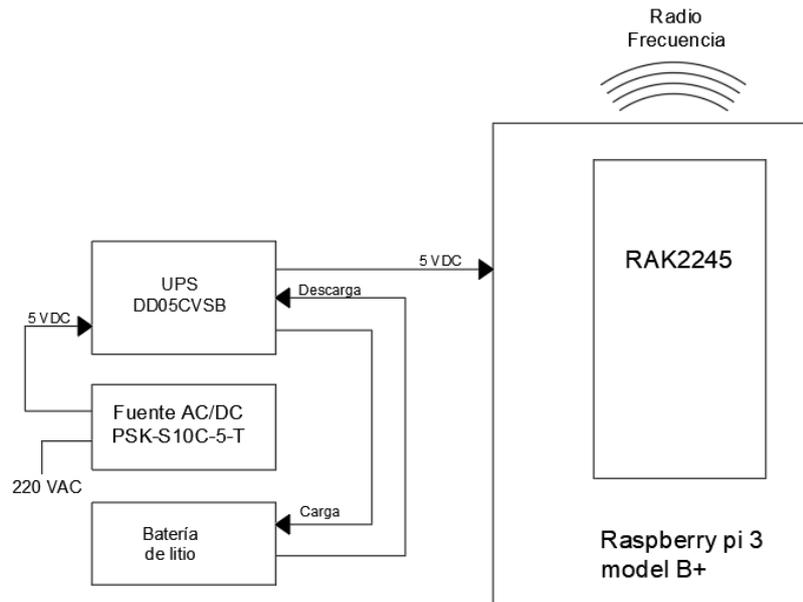


Figura 4.2 Arquitectura de hardware Gateway

4.2.2 Modelo 3D del sistema

En las figuras 4.3 y 4.4 se muestran los modelos 3D finales del Nodo Sensor y del Gateway. En ellos se puede apreciar la distribución de los componentes mostrados en la arquitectura, los cuales se encuentran montados en las PCBs mostradas en el anexo 2.

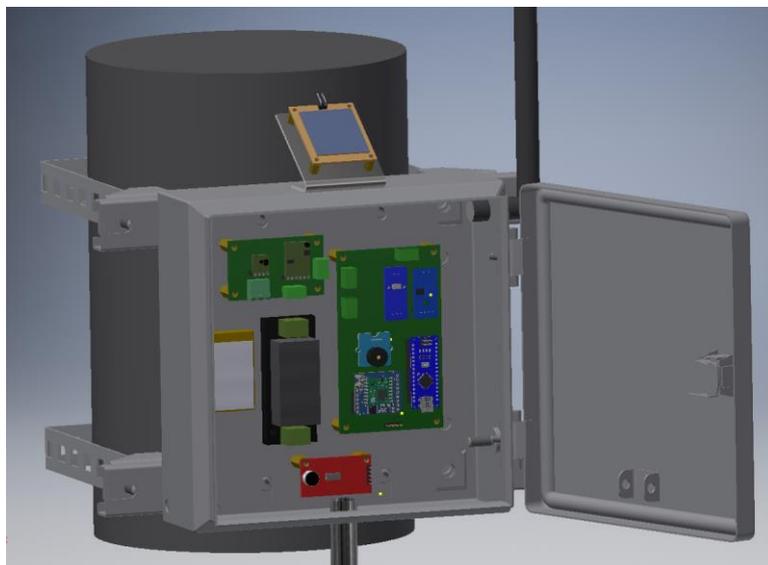


Figura 4.3 Modelo 3D Nodo Sensor

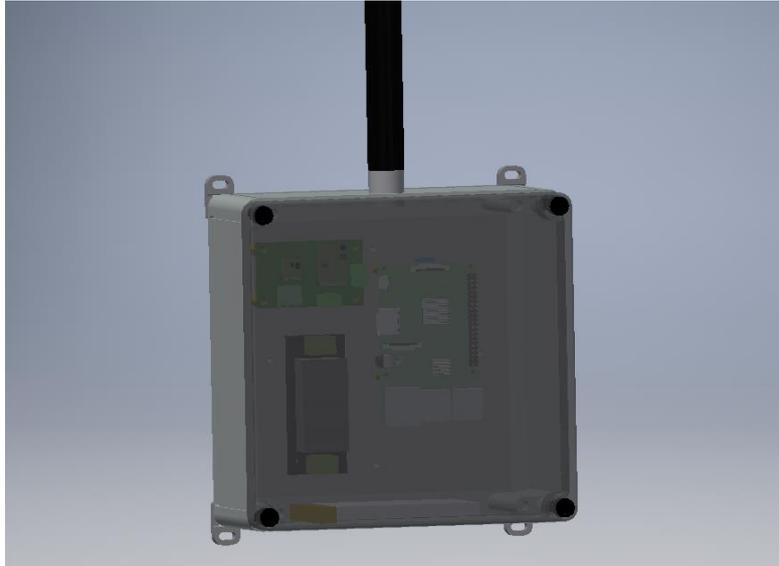


Figura 4.4 Modelo 3D Gateway

4.2.3. Selección de componentes electrónicos y de procesamiento

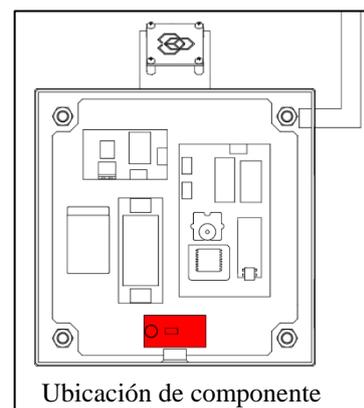
En esta sección se seleccionarán los componentes electrónicos y de procesamiento del Nodo Sensor y el Gateway basado en la arquitectura de hardware y los requerimientos del sistema, la ubicación de los componentes en el sistema estará marcada en rojo a través de un gráfico al inicio de cada sección.

4.2.3.1. Nodo Sensor:

Sensor de sonido:

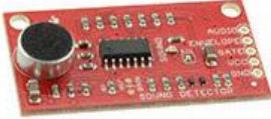
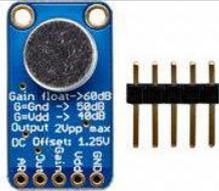
Para la medición de la contaminación acústica se necesita un sensor de sonido con las siguientes características:

- Bajo consumo energético
- Energización por microcontrolador (3.3V-5V)
- Buena precisión en la salida analógica



En la tabla 4.1 se presentan 3 alternativas de sensores de sonido que cumplen con las características mencionadas.

Tabla 4.1 Sensores de sonido

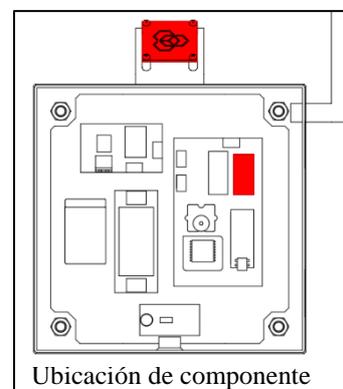
Proveedor	Digikey		
Fabricante	SparkFun Electronics	Adafruit Industries LLC	Waveshare
Modelo	 SEN-12642	 MAX9814	
Dimensiones en mm (largo x ancho)	44 x 23		38.42 x 18.67
Entrada	3.5V-5V	2.7V-5.5V	3.3V-5.3V
Ganancias	6-60dB	40-60dB	
Rango de frecuencia	20Hz ~ 20kHz	20Hz ~ 20kHz	50Hz ~ 20KHz
Sensibilidad	-44 dB \pm 2 dB	-44 dB \pm 2 dB	-44 dB \pm 2 dB
Rango de temperatura de trabajo		-20°C a +85°C	
Data analógica	Relación entre voltaje y amplitud de onda real	Relación entre voltaje y nivel de presión sonora (inexacto)	Relación entre voltaje y nivel de presión sonora (inexacto)
Precio	\$10.95	\$7.95	\$3.89

Se optará por el modelo SEN-12642 debido a que posee una mayor precisión en la data analógica que entrega al microcontrolador y porque tiene un rango mayor de ganancias lo cual permitirá una mejor calibración en función a la posición del sensor.

Sensor de lluvia:

Para la detección de lluvia se utilizará un sensor que cumpla con las siguientes características:

- Bajo consumo energético
- Energización por el microcontrolador (3.3V-5V)



En la tabla 4.2 se presentan 2 alternativas de sensores de lluvia que cumplen con las características mencionadas.

Tabla 4.2 Sensores de lluvia

Proveedor	Amazon	
Fabricante	AMG kits	Solu
Modelo	 FC-37	 YL-83

Dimensiones en mm (largo x ancho)	32 x 15 (PCB) 50x40 (Celda)	32 x 14 (PCB) 50x40 (Celda)
Entrada	3.5V-5V	3.5V-5V
Corriente máxima	15mA	15mA
Rango de temperatura de trabajo	-20°C a +85°C	-20°C a +85°C
Data analógica	Señal analógica proporcional a la cantidad de agua en la celda	Señal analógica proporcional a la cantidad de agua en la celda
Precio	\$1.99	\$3.21

En este caso ambos dispositivos poseen las mismas características y lo único que varía es la marca, debido a esto se seleccionará el modelo FC-37 por su menor precio.

Sensor de corriente:

Para la detección de pérdidas de energía el nodo sensor dispondrá de un sensor de corriente que informará cuando haya un corte de energía (pasando a funcionar las baterías). Para esto se ha seleccionado el sensor ACS712-05 el cual posee las características mostradas en la tabla 4.3.

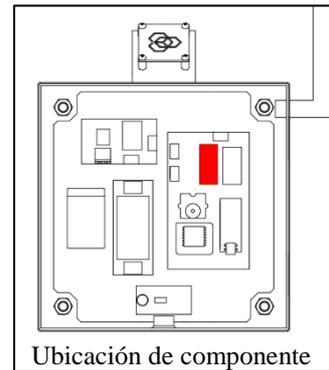


Tabla 4.3 Sensor de corriente

Proveedor	Aliexpress
Fabricante	Allegro
Modelo	 ACS712-05
Dimensiones en mm (largo x ancho)	32 x 14 (aproximadas)
Entrada	5V
Corriente máxima	10mA
Rango de temperatura de trabajo	-40°C a +85°C
Data analógica	Señal analógica proporcional a la cantidad de corriente entre -5 y 5 A
Precio	\$1.06

Controlador:

Durante el diseño conceptual de la solución se propuso el uso de un microcontrolador Arduino para el control del nodo sensor, se seleccionó el modelo nano por su tamaño compacto y por su capacidad de operar con los sensores seleccionados. En la tabla 4.4 se presentan sus principales características.

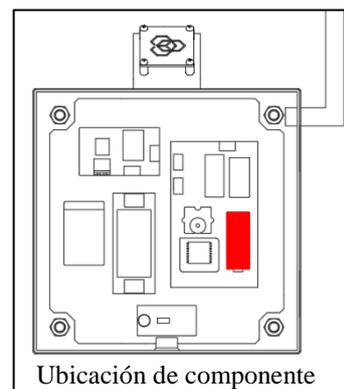


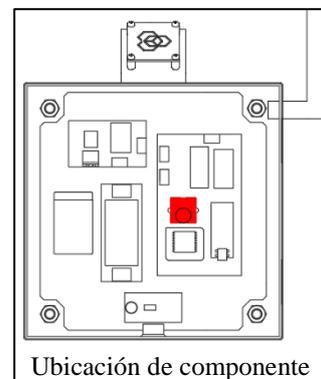
Tabla 4.4 Microcontrolador Nodo Sensor

Proveedor	Naylamp Mechatronics
Fabricante	Arduino
Modelo	 ARD-NANOCH
Microcontrolador	ATmega328P
Arquitectura	AVR
Voltaje de operación	5 V
Memoria Flash	32 KB donde 2 KB son usados por el bootloader
SRAM	2 KB
Velocidad del clock	16 MHz
Pines analógicos	8
EEPROM	1 KB
Corriente por pin	40 mA (I/O Pins)
Voltaje de alimentación	7-12 V
Digital I/O Pins	22 (6 of which are PWM)
PWM Output	6
Consumo energetico	19 mA
Tamaño de PCB	18 x 45 mm
Peso	7 g
Precio	\$6.67

Buzzer:

Para la revisión del estado de los sensores de sonido se utilizará un actuador buzzer que cumpla con las siguientes características:

- Bajo consumo energético
- Energización por el microcontrolador (3.3V-5V)



En la tabla 4.6 se presentan 2 alternativas que cumplen con las características mencionadas.

Tabla 4.6 Módulos Buzzer

Proveedor	Digikey	
Fabricante	DFRobot	Groove
Modelo	 Digital Buzzer Module	 Groove Buzzer

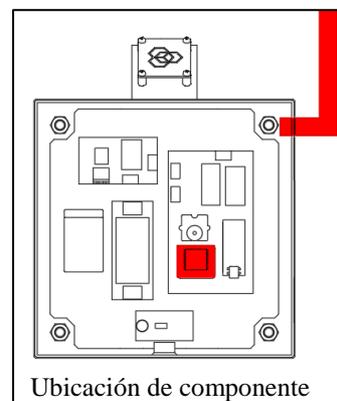
Dimensiones en mm (largo x ancho)	20 x 30 (aproximadas)	20 x 20 (aproximadas)
Entrada	3.5V-5V	3.5V-5V
Corriente máxima	50mA	50mA
Rango de temperatura de trabajo	-20°C a +70°C	-20°C a +70°C
Salida de sonido	≥ 85dB	≥ 85dB
Frecuencia de resonancia	2300±300Hz	2300±300Hz
Precio	\$4.23	\$2.07

En este caso ambos dispositivos poseen las mismas características principales, el modelo de DFRobot permite un mejor control en la variación de tonalidades sin embargo en este proyecto el buzzer servirá únicamente de prueba para los sensores de sonido por lo que se seleccionará el modelo de Grove debido a su menor precio y tamaño.

Transceptor LoRa:

Para la comunicación entre el nodo sensor y el Gateway se utilizarán dispositivos compatibles con la señal de radiofrecuencia que utiliza la tecnología LoRa y que además posean las siguientes características:

- Bajo consumo energético
- Energización por el microcontrolador (3.3V-5V)



En la tabla 4.7 se presentan 2 alternativas que cumplen con las características mencionadas.

Tabla 4.7 Transceptores LoRa

Proveedor	Digkey	
Fabricante	RFM Solutions	Murata Electronics
Modelo	 RFM95W-915S2	 CMWX1ZZABZ-093
Dimensiones en mm (largo x ancho)	16 x 16 x 1.8	-
Voltaje de entrada	1.8V-3.7V	3.3V
Corriente máxima	20mA-120mA	106mA
Rango de temperatura de trabajo	-20°C a +70°C	-40°C a +85°C
Frecuencia de transmisión	915 MHz	868MHz-915MHz
Precio	\$13.84	\$17.82

Según el cuadro comparativo, el modelo CMWX1ZZABZ-093 posee mayor flexibilidad en cuanto a frecuencias. Sin embargo, las características del modelo RFM95W-915S2 cumplen con el requerimiento y tiene un menor precio por lo que será el seleccionado.

El transceptor seleccionado requerirá de una antena que transmita a la frecuencia del sistema (915 MHz), en la tabla 4.8 se muestra la antena seleccionada.

Tabla 4.8 Antena Nodo Sensor

Proveedor	SparkFun
Fabricante	ChangHong
Modelo	 WRL-14876
Dimensiones en mm (largo x diámetro)	200 x 13.7
Frecuencia de transmisión	915 MHz
Precio	\$7.95

4.2.3.2. Gateway

Controlador:

Durante el diseño conceptual de la solución se propuso el uso de un microprocesador Raspberry pi para el control del Gateway, se seleccionó el modelo 3 B+ por ser el más moderno y tener mejor rendimiento que sus predecesores. En la tabla 4.5 se presentan sus principales características.

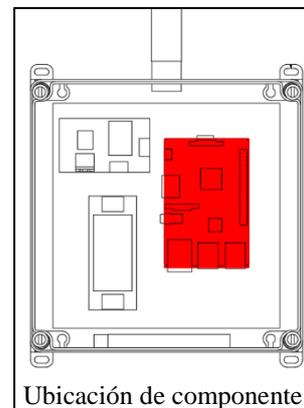


Tabla 4.5 Microcontrolador Gateway

Proveedor	CanalKit
Fabricante	Raspberry Pi
Modelo	 Raspberry Pi 3 Model B+
Procesador	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
RAM	1GB LPDDR2 SDRAM
Wireless	2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE

Ethernet	Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
GPIO	Extended 40-pin GPIO header
HDMI	Full-size HDMI
USB	4 USB 2.0 ports
Video	4-pole stereo output and composite video port
SD	Micro SD port for loading your operating system and storing data
Alimentación	5V/2.5A DC power input
Precio	\$35

La Raspberry pi 3 posee un consumo de corriente promedio sin periféricos de 500mA

Transceptor LoRa:

El Gateway utilizara como transceptor el módulo RAK2245, el cual es compatible con Raspberry Pi y permite una fácil configuración como LoRa Gateway de hasta 8 canales en simultáneo. En la tabla 4.9 se muestra una descripción más detallada del módulo:

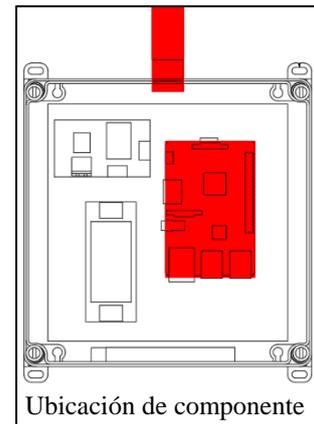


Tabla 4.9 Modulo RAK2245

Proveedor	Aliespress
Fabricante	RAK
Modelo	<p>WiLink LoRa Board RAK2245 Pi HAT NEW RAK2245</p>
Dimensiones en mm (largo x ancho x alto)	56 x 30 x 22
Voltaje de entrada	5V
Corriente máxima	360mA
Rango de temperatura	-40°C a +85°C
Frecuencia de transmisión	Todas las frecuencias LoRa
Precio	\$120

Este módulo requiere una antena de mayor capacidad que los nodos sensores ya que deberá recibir y transmitir múltiples señales en simultáneo, en la tabla 4.10 se muestra la antena seleccionada.

Tabla 4.10 Antena Nodo Sensor

Proveedor	Aliespress
Fabricante	RAK
Modelo	 RAK Antenna
Dimensiones en mm (largo x diametro)	400 x 20
Frecuencia de transmisión	915 MHz
Precio	\$27.31

4.2.4. Selección de componentes de energía y potencia

En esta sección se seleccionarán los componentes de energía y potencia para lo cual primero se realizarán cálculos energéticos en función a los componentes seleccionados previamente. Al igual que en la sección anterior, un gráfico mostrará la ubicación de los componentes en el sistema.

4.2.4.1. Cálculos previos

La energía de alimentación tanto del Nodo sensor como del Gateway será a través de fuentes primarias y secundarias:

- Una fuente primaria de corriente alterna 220VAC-60Hz
- Una fuente secundaria de baterías de litio 3.7V cargada por la fuente primaria

Antes de seleccionar correctamente los componentes de energía y potencia se calculará el consumo máximo de potencia de los componentes seleccionados para el nodo sensor y el Gateway como se aprecia en la tabla 4.11:

Tabla 4.11 Consumo máximo de potencia de componentes seleccionados

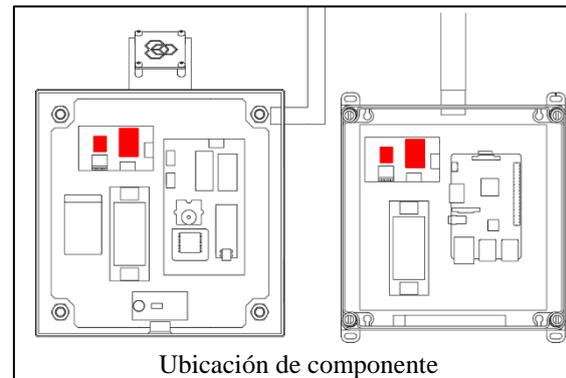
Elemento	Componente	Voltaje (V)	Corriente (A)	Potencia (W)
Nodo sensor	SEN-12642	5.0	0.034	0.170
	FC-37	5.0	0.015	0.075
	ACS712-05	5.0	0.010	0.050
	Arduino Nano	5.0	0.019	0.095
	Groove Buzzer	5.0	0.050	0.250
	RFM95W-915S2	3.3	0.120	0.400
Gateway	Raspberry pi3	5.0	0.500	2.500
	RAK2245	5.0	0.360	1.800

Según la tabla, la potencia total requerida por el Nodo sensor es de **1.5W** y la potencia total requerida por el Gateway es de **4.3W**. Agregando un factor de seguridad de **1.2**, se

seleccionarán componentes que suministren potencias de al menos **1.8W** para el nodo sensor y **5.1W** para el Gateway.

4.2.4.2. Componentes de potencia

Como se mencionó en la descripción de la solución 1, se utilizarán placas UPS para poder asegurar un suministro estable de energía a pesar de que se produzcan cortes en la fuente primaria tanto para el Nodo Sensor como para el Gateway. Ya que estas placas serán las que finalmente alimentarán a los componentes del sistema, serán las primeras en ser seleccionadas.



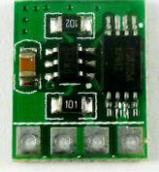
En la tabla 4.12 se presentan 2 alternativas de placas UPS que cumplen con los requerimientos de potencia de 1.8W del nodo sensor:

Tabla 4.12 Selección de placa UPS

Proveedor Fabricante	Mouser Electronics Sola / Hevi-Duty	Aliexpress Diy
Modelo	 SDUPMBRK	 DD05CVSB_5V
Dimensiones en mm (largo x ancho x alto)	-	23 x 15.4 x 5.7
Voltaje de entrada	22.5V-30V	4.5V-8V
Voltaje de salida	24V	5V
Voltaje de batería	24V	3.7V
Potencia Máxima	240W	6W
Tipo de batería	Interna	Lithium Ion/ Lithium Polymer
Proteccion contra sobre carga	Si	No
Peso	220g	1.8g
Precio	\$24.49	\$3.18

La primera opción es la fuente UPS industrial de menor potencia y precio en el catálogo, la cual se puede ver que sobrepasa la cantidad de potencia consumida por el Nodo Sensor, debido a esto, se seleccionará el componente DD05CVSB_5V el cual es una placa electrónica a la que se le conecta una fuente de alimentación y una batería y esta se encarga de suministrar la energía y mantener la batería cargada. Además, para solucionar el problema sobrecarga se utilizará el siguiente componente protector recomendado por el fabricante el cual se muestra en la tabla 4.13:

Tabla 4.13 Protección contra sobrecarga

Proveedor	Aliexpress
Fabricante	Diy
Modelo	 DD05CVSA
Dimensiones en mm (largo x ancho x alto)	12.6 x 10.2 x 2.1
Baterías compatibles	3.7V/4.2V
Protecciones	Sobrecarga Sobredescarga Corto Circuito Sobrecorriente
Precio	\$1.89

4.2.4.3. Componentes de energía

Para determinar la potencia que debe suministrar la fuente de alimentación y el tipo de batería que se utilizara es necesario hacer un cálculo adicional. Para esto se tomarán en cuenta los siguientes datos de la placa UPS seleccionada ya que esta se encargará de recibir la energía de la fuente de alimentación y utilizarla para energizar los dispositivos y recargar las baterías:

- Voltaje de carga recomendado: 5V
- Corriente de carga de batería: 0A-1A
- Corriente de descarga de batería: 0A-2A
- Corriente de salida: 0A-1.2A
- Voltaje de salida: 5V
- Eficiencia de conversión de descarga: 78%-90%

Nodo Sensor:

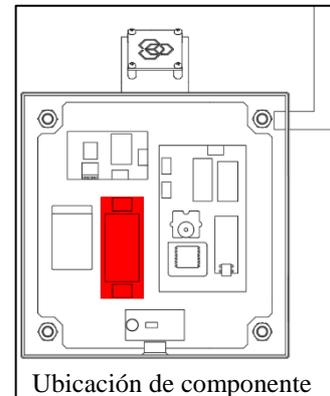
Con los datos mostrados y asumiendo una eficiencia de conversión crítica del 78% se determinaron los siguientes parámetros para la fuente del NS:

- **Potencia Fuente NS:**

$$\text{Potencia del NS} \div \text{Eficiencia de conversión} \rightarrow 1.8 \div 0.78 = 2.30W$$

- **Corriente:**

$$\text{Potencia Fuente} \div \text{Voltaje recomendado} \rightarrow 2.3 \div 5 = 0.46A$$



En la tabla 4.14 se muestra la fuente de energía seleccionada que cumple con las características mencionadas:

Tabla 4.14 Fuente de alimentación Nodo Sensor

Proveedor	Mouser Electronics
Fabricante	CUI INC
Modelo	 PSK-S6C-5-T
Dimensiones en mm (largo x ancho x alto)	76.0 x 31.5 x 24.16
Entrada	85VAC-264VAC
Salida	5VDC/1.2A
Potencia	6W
Rango de temperatura de trabajo	-40°C a +70°C
Eficiencia	76%
Peso	52g
Precio	\$17.11

Ahora se calculará la capacidad de la batería de ion de litio para una duración promedio de 12 horas sin el suministro primario de energía, para esto se considerará que en modo batería no se utilizará el Buzzer y el sensor de lluvia rara vez funcionará a su máxima capacidad. Además, la transmisión de datos consume un máximo de 120mA, pero ocurre 1 vez por minuto y en fracciones de segundo por lo que no se considerará en el cálculo:

- **Potencia Modo batería:**

Potencia consumida ÷ Eficiencia de conversión

$$\rightarrow 0.315 \div 0.78 = 0.40W$$

- **Corriente que suministrará la batería:**

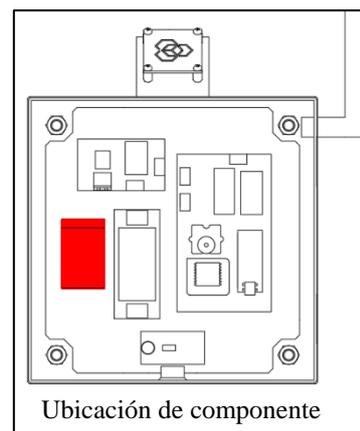
Potencia Modo ÷ Batería Voltaje de batería

$$\rightarrow 0.40 \div 3.7 = 0.1A = 100mA$$

- **Capacidad requerida por la batería:**

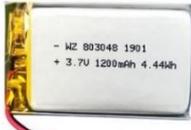
Corriente que suministrará la batería × Cantidad

$$\text{de Horas} \rightarrow 100 \times 12 = 1200mAh$$



Se requiere una batería de al menos **1200mAh** para mantener el nodo sensor funcionando por 12 horas. En la tabla 4.15 se muestra la batería seleccionada

Tabla 4.15 Batería de ion de litio Nodo sensor

Proveedor	Alibaba
Modelo	 <p>803048</p>
Dimensiones en mm (largo x ancho x alto)	8 x 30 x 48
Voltaje de carga	4.2V
Voltaje de salida	3.7V
Capacidad	1200mAh
Recargable	Si
Rango de temperatura de trabajo	-20°C a +50°C
Peso	10g
Precio	\$1.60

Gateway:

Con los datos mostrados y asumiendo una eficiencia de conversión crítica del 78% se determinaron los siguientes parámetros para la fuente del Gateway:

- **Potencia Fuente Gateway:**

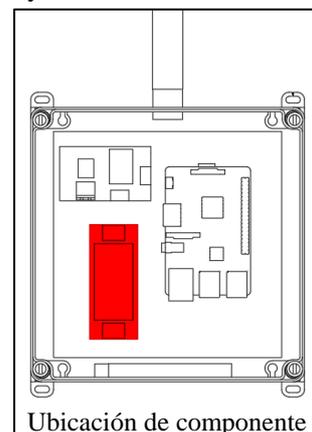
Potencia del Gateway ÷ Eficiencia de conversión →

$$5.1 \div 0.78 = 6.54W$$

- **Corriente:**

Potencia Fuente ÷ Voltaje recomendado → $6.54 \div 5 =$

$$1.3A$$



En la tabla 4.16 se muestran el componente seleccionado que cumple con las características mencionadas:

Tabla 4.16 Fuente de alimentación Gateway

Proveedor	Mouser Electronics
Fabricante	CUI INC
Modelo	 PSK-S10C-5-T
Dimensiones en mm (largo x ancho x alto)	76.0 x 31.5 x 27.8
Entrada	85VAC-264VAC
Salida	5VDC/2A
Potencia	10W
Rango de temperatura de trabajo	-40°C a +70°C
Eficiencia	76%
Peso	68g
Precio	\$17.11

Ahora se calculará la capacidad de la batería de litio para una duración máxima de 4 horas sin el suministro primario de energía, solo se considerarán 4 horas debido al notable mayor consumo energético de los Gateway y la fácil accesibilidad que tendrán para solucionar problemas energéticos al estar ubicados en paredes de edificios.

- **Potencia Modo batería:**

Potencia consumida ÷ Eficiencia de conversión

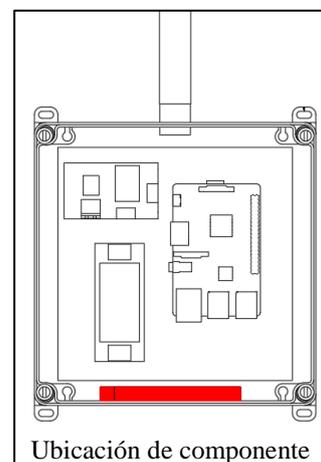
$$\rightarrow 4.3 \div 0.78 = 5.51W$$

- **Corriente que suministrará la batería:**

$$\text{Potencia Modo Batería} \div \text{Voltaje de batería} \rightarrow 5.51 \div 3.7 = 1.48A = 1480mA$$

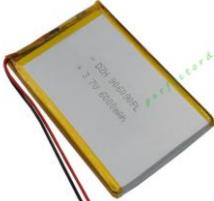
- **Capacidad requerida por la batería:**

$$\text{Corriente que suministrará la batería} \times \text{Cantidad de Horas} \rightarrow 1480 \times 4 = 5920mAh$$



Se requiere una batería de al menos **5920mAh** para mantener el Gateway funcionando por 4 horas. En la tabla 4.17 se muestra la batería seleccionada

Tabla 4.17 Batería de ion de litio Gateway

Proveedor	Ebay
Modelo	 906090
Dimensiones en mm (largo x ancho x alto)	9 x 60 x 90
Voltaje de carga	4.2V
Voltaje de salida	3.7V
Capacidad	6000mAh
Recargable	Si
Rango de temperatura de trabajo	-10°C a +45°C
Peso	108g
Precio	\$12.00

4.2.5. Selección de componentes mecánicos

4.2.5.1. Protección y sujeción

Nodo Sensor:

Para la selección de la caja Rittal donde ira ubicada toda la electrónica y control del nodo sensor se debe tomar en cuenta las siguientes características:

- La caja debe tener una protección contra el agua y polvo IP65 o más.
- Debe tener una apertura sencilla para el mantenimiento.
- Debe tener suficiente espacio para acomodar los componentes internos.
- Debe poder instalarse en postes de alumbrado, teléfono o eléctrico.

Lo primero que se debe definir son las dimensiones de la caja, para esto se realizó un diagrama de distribución de los componentes electrónicos con las dimensiones obtenidas de la selección de componentes. En la figura 4.3 se aprecia el diagrama.

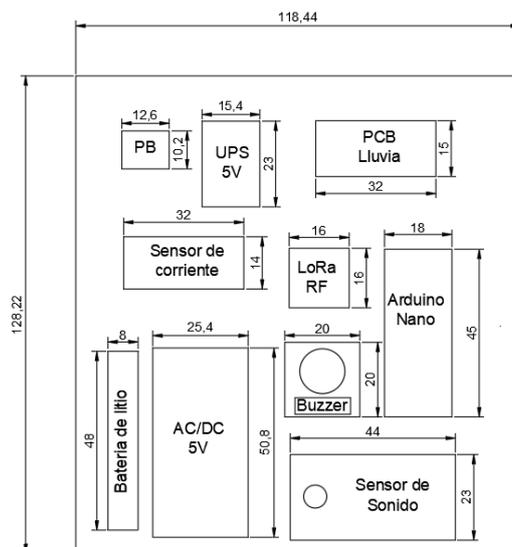


Figura 4.5 Diagrama de distribución de componentes Nodo Sensor

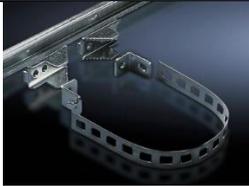
Con el diagrama se definieron dimensiones mínimas de 130 mm x 120 mm de área y un espesor mínimo ya que en general los componentes son bastante delgados. En la tabla 4.18 se muestra el componente seleccionado que cumple con las características mencionadas.

Tabla 4.18 Caja Rittal Nodo Sensor

Proveedor Fabricante	Automation24 Rittal – The System
Modelo	 EB 1546.500
Dimensiones de caja en mm (largo x ancho x profundidad)	200 x 200 x 80
Dimensiones de placa de montaje en mm (largo x ancho x espesor)	185 x 175 x 2
Material	Chapa de acero
Color	RAL 7035
Grado de protección	IP66
Peso	2,4 kg
Precio	\$37.72

La caja Rittal seleccionada cuenta con una placa de montaje con las dimensiones requeridas, además tiene protección IP66 la cual le da inmunidad al polvo y resistencia a agua arrojada a chorros. Para el montaje en los postes se utilizará un accesorio de fijación compatible el cual se muestra en la tabla 4.19.

Tabla 4.19 Elemento de sujeción Nodo sensor

Proveedor	Automation24
Fabricante	Rittal – The System
Modelo	 SZ 2584.000
Diámetros aceptados en mm	40 mm a 190 mm
Material de perfil	Chapa de acero
Material de brida de fleje	Acero inoxidable
Peso	2 kg
Precio	\$37.63

Por seguridad se calculará el diámetro mínimo de los pernos de sujeción que deben sostener la caja y se comparará con los proporcionados en el producto. Para este cálculo se sabe que el sistema de sujeción utilizara 4 agujeros ubicados en las esquinas posteriores de la caja para colocar los pernos.

La información obtenida de las masas significativas de los componentes seleccionados se muestra en la tabla 4.20. Con estos datos se calculará la fuerza a la que estarán sometidas los pernos debido a los componentes del sistema.

Tabla 4.20 Masa y Peso de los componentes seleccionados

Componente	Masa (kg)	Peso (N)
Arduino Nano	0.007	00.07
Fuente AC/DC	0.052	00.51
Batería de litio	0.010	00.10
Caja Rittal	2.4	23.54
Base de sensor de lluvia	0.072	00.71
Total	2.54	24.92

Adicionalmente, de acuerdo al artículo 12 de la Norma Técnica Peruana de Edificación (Norma E-020) se calcularán las cargas debido al viento en la caja del nodo sensor. Primero se calculará la velocidad de diseño del viento según los parámetros mostrados en la tabla 4.21.

Tabla 4.21 Parámetros para velocidad de diseño

Parámetro	Valor	Justificación
Clasificación de la estructura	Tipo 1	Estructura poco sensible a las ráfagas de viento debido a su forma y dimensiones
Velocidad de diseño hasta 10 m de altura en Km/h	75 km/h	La velocidad de viento más alta en Lima y Callao según el atlas eólico es del 13 km/h, sin embargo la norma indica tomar como valor mínimo 75 km/h
Altura sobre el terreno	4 m	La altura de colocación de los nodos sensores será de aproximada mente 4 metros

Siguiendo la norma E-020 se aplicará la siguiente fórmula para calcular la velocidad de diseño del viento:

$$V_h = V \cdot (h/10)^{0.22}.$$

donde:

- V_h es la velocidad de diseño del viento a una altura h
- V es la Velocidad de diseño hasta 10 m de altura
- h es la altura sobre el terreno

Con los parámetros previamente definidos se obtuvo una velocidad de diseño de **61.31 km/h.**

Finalmente se calculará la presión del viento para la cual se utilizará un factor de forma de 2 debido a la estructura rectangular de la caja y siguiendo la tabla de la norma E.020 utilizando la siguiente formula:

$$P_h = 0,005 \cdot C V_h^2.$$

donde:

- P_h es la presión del viento a una altura h .
- V_h es la velocidad de diseño del viento a una altura h
- C es el factor de forma

El cálculo dio como resultado una presión de 37.6 Kgf/m² o **368.86 N/m²** y ya que el área más crítica para los pernos debido a la fuerza del viento es de 0.0164 m² (área lateral), la fuerza que se genera sobre la caja sería de:

$$F = 368.86 \times 0.0164 = 6N$$

En la figura 4.4 se muestra el diagrama de fuerzas con los resultados obtenidos y el diseño de la caja.

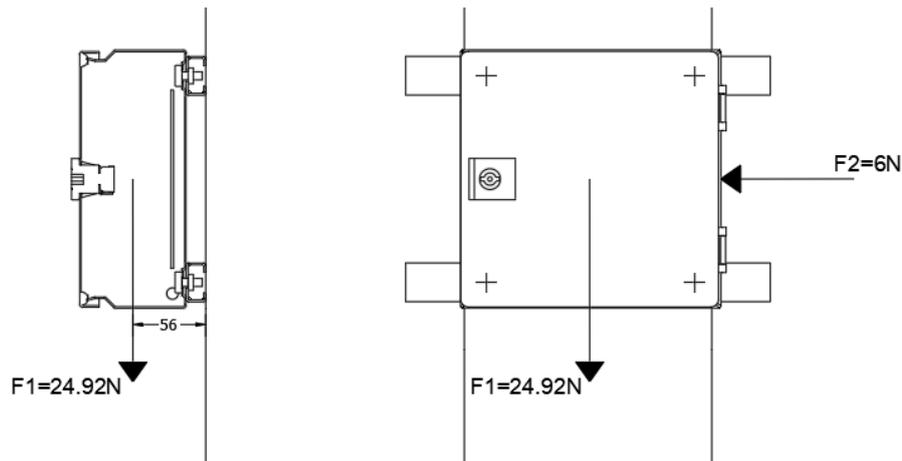


Figura 4.6 Diagrama de fuerzas del Nodo Sensor

Siguiendo con el cálculo de los pernos de sujeción de la caja, la fuerza resultante en uno de los pernos (F_p) saldría de la siguiente ecuación:

$$F_p = \sqrt{(F1/4)^2 + (F2/4)^2} = 6.4N.$$

Ya que el esfuerzo normal que se podría generar en los pernos es despreciable debido a las dimensiones de la caja, se calculará el diámetro mínimo de los pernos debido al esfuerzo cortante que generan las fuerzas halladas. Para este cálculo se considerará que el esfuerzo cortante máximo que pueden soportar los pernos del tipo A325 es de aproximadamente 115Mpa:

$$115 = \frac{F_p}{\pi \cdot (d_m/2)^2}.$$

De la ecuación anterior se obtiene que el diámetro mínimo de los pernos es de **0.27 mm** y ya que el soporte utiliza pernos **M8** se puede afirmar que estos no fallaran por corte.

Gateway:

Para la selección de la caja Rittal donde ira ubicada toda la electrónica y control del Gateway se debe tomar en cuenta las siguientes características:

- El interior de la caja debe estar protegido contra el agua y el polvo.
- Debe tener una apertura sencilla para el mantenimiento.
- Debe tener suficiente espacio para acomodar los componentes internos.
- Debe poder instalarse en paredes de ladrillo o drywall.

Al igual que para el nodo sensor se realizó un diagrama de distribución de los componentes electrónicos con las dimensiones obtenidas de la selección de componentes. En la figura 4.5 se aprecia el diagrama.

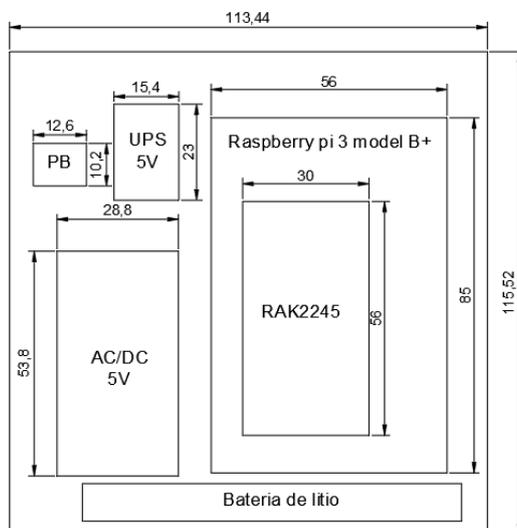


Figura 4.7 Diagrama de distribución de componentes Gateway

Con el diagrama se definieron dimensiones mínimas de 114 mm x 116 mm de área y un espesor mínimo de 60mm debido a la posición de la batería.

En la tabla 4.22 se muestran el componente seleccionado que cumple con las características mencionadas:

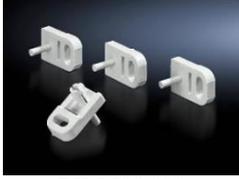
Tabla 4.22 Caja Rittal Gateway

Proveedor	Transfer Multisort Elektronik
Fabricante	Rittal – The System
Modelo	 PK 9517.100
Dimensiones de caja en mm (largo x ancho x profundidad)	182 x 180 x 90
Material	Policarbonato reforzado con fibra de vidrio
Color	RAL 7035
Grado de protección	IP66
Peso	0.6 kg
Precio	\$30.86

La caja rittal seleccionada tiene protección IP66 la cual le da inmunidad al polvo y

resistencia a agua arrojada a chorros. Para el montaje en la pared se utilizará un accesorio de fijación compatible el cual se muestra en la tabla 4.23.

Tabla 4.23 Elemento de sujeción Gateway

Proveedor	Transfer Multisort Elektronik
Fabricante	Rittal – The System
Modelo	 PK 9583.000
Aplicación	Todas las cajas PK
Material	Poliamida
Color	RAL 7035
Peso	0.16 kg
Precio	\$1.22

Ya que el peso de los componentes del Gateway es menor a los del Nodo Sensor, se omitirá el cálculo de los pernos de sujeción.

4.2.5.2. Protección de micrófono

Nodo Sensor

Para la protección del micrófono se ha seleccionado una espuma cortavientos estándar para sonómetros, en la tabla 4.24 se pueden ver sus características principales.

Tabla 4.24 Elemento protector de micrófono

Proveedor	Amazon
Fabricante	REED Instruments
Modelo	 SB-01
Diámetro de agujero	0.5"
Peso	18 g
Precio	\$5.69

4.2.6. Diagramas de conexiones:

En las figuras 4.6 y 4.7 se muestran los diagramas de conexiones tanto del Gateway como del Nodo Sensor. Estos diagramas muestran detalladamente todas las conexiones del sistema, sin embargo, para facilitar el montaje se han diseñado 2 PCBs que ya cuenten con la mayoría de las conexiones mostradas y solo se deberán conectar los componentes

a través de terminales hembra. Para visualizar esquemáticos y conexiones de las PCBs al sistema revisar anexos 2 y 3.

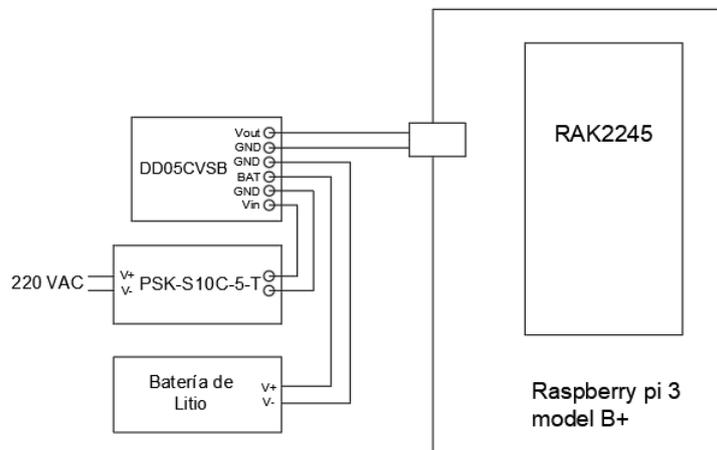


Figura 4.8 Diagrama de conexiones Gateway

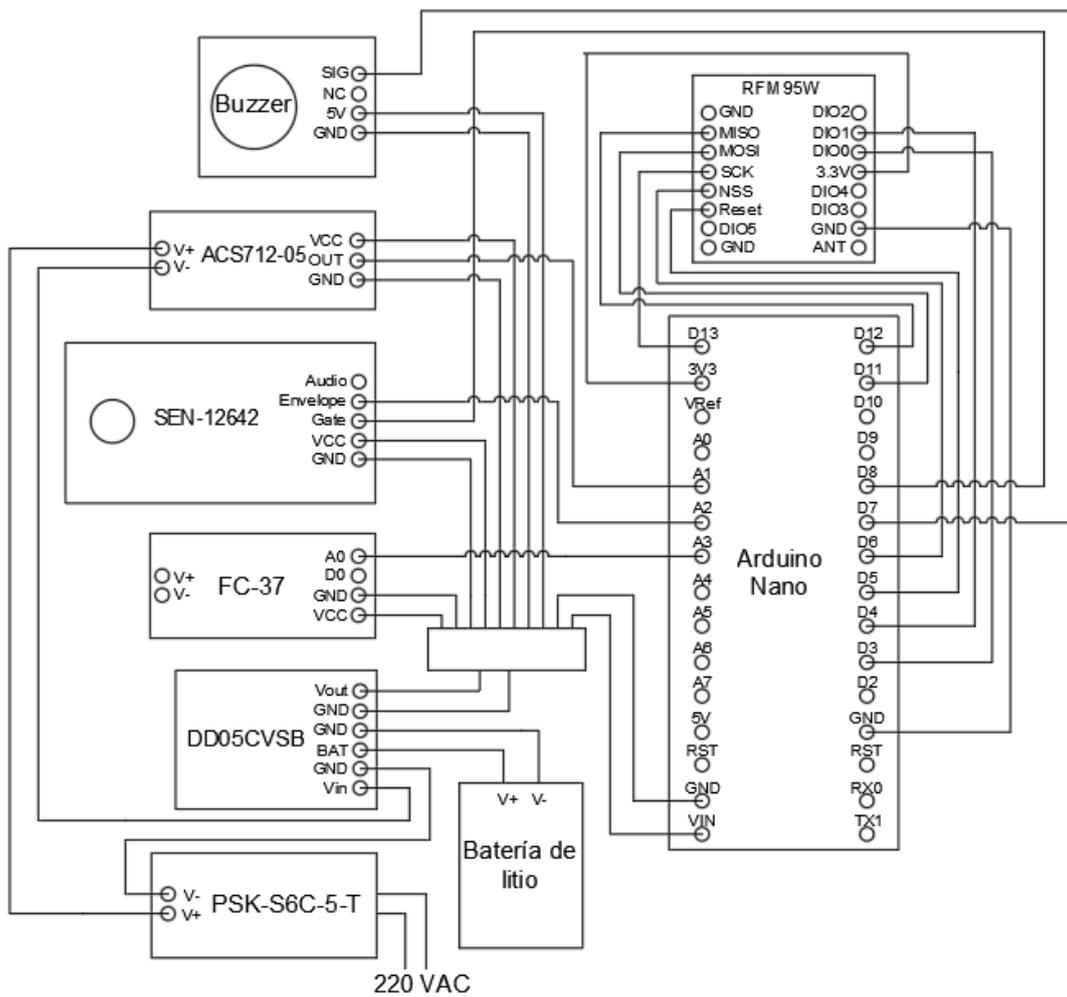


Figura 4.9 Diagrama de conexiones Nodo Sensor

4.2.7. Diagramas de flujo:

En esta sección se detallarán los diagramas de flujo de los nodos sensores donde se comprenderá mejor el funcionamiento desde la toma de datos hasta el envío de los mismo al sistema ciber. No se verá un diagrama de flujo de los Gateway ya que serán considerados parte del sistema ciber porque siguen una programación preestablecida por el servidor TTN donde captan cualquier señal de radiofrecuencia que cumpla con su protocolo y claves de acceso y los suben al servidor a través de internet.

4.2.7.1. Diagrama general de nodo sensor

En la figura 4.8 se muestra el diagrama de flujo general del controlador del nodo sensor, en este se puede apreciar de forma resumida el proceso de verificación y obtención de datos para finalmente enviar solo la información necesaria al sistema ciber minimizando el consumo de energía y el flujo de datos.

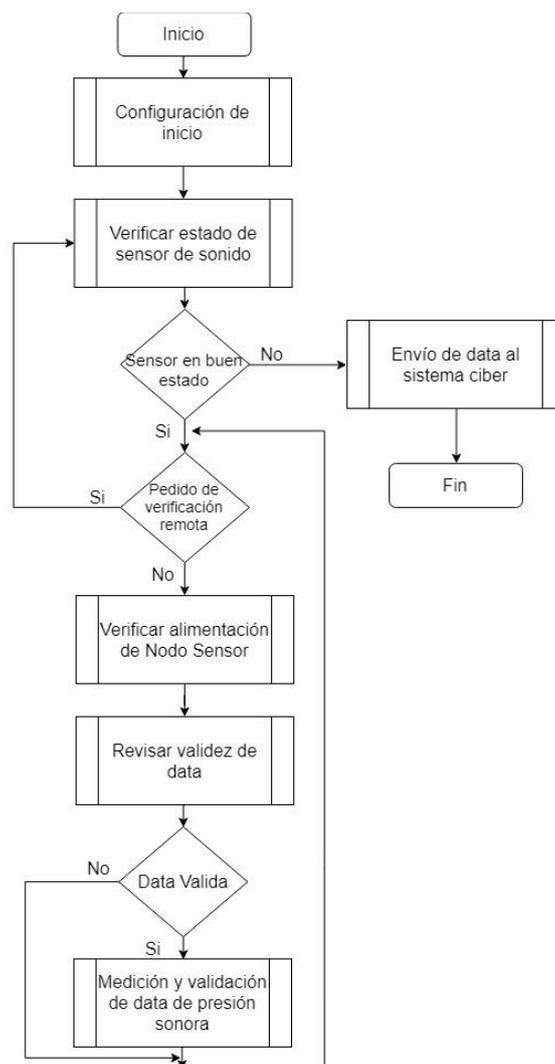




Figura 4.10 Diagrama de flujo general

4.2.7.2. Configuración de inicio

Antes de iniciar las tareas se deben cargar las librerías, configurar los puertos y declarar las variables a utilizar, también se realizará la configuración del transceptor LoRa para asegurar un correcto envío de la data. En la figura 4.9 se muestra este proceso.

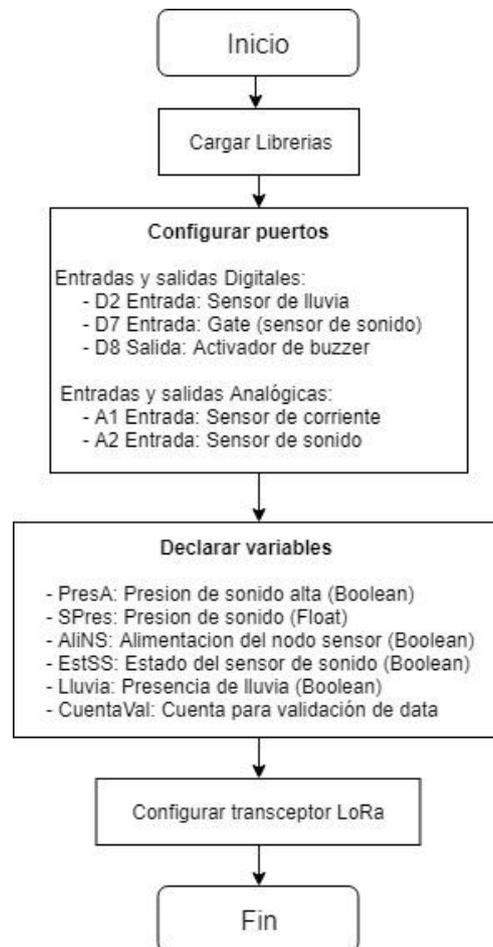


Figura 4.11 Diagrama de flujo de configuración inicial

4.2.7.3. Verificar estado de sensor de sonido

Esta función se encargará de asegurar que el sensor de sonido funciona de manera correcta a través de su testeo con el buzzer. En la figura 4.10 se muestra este proceso.

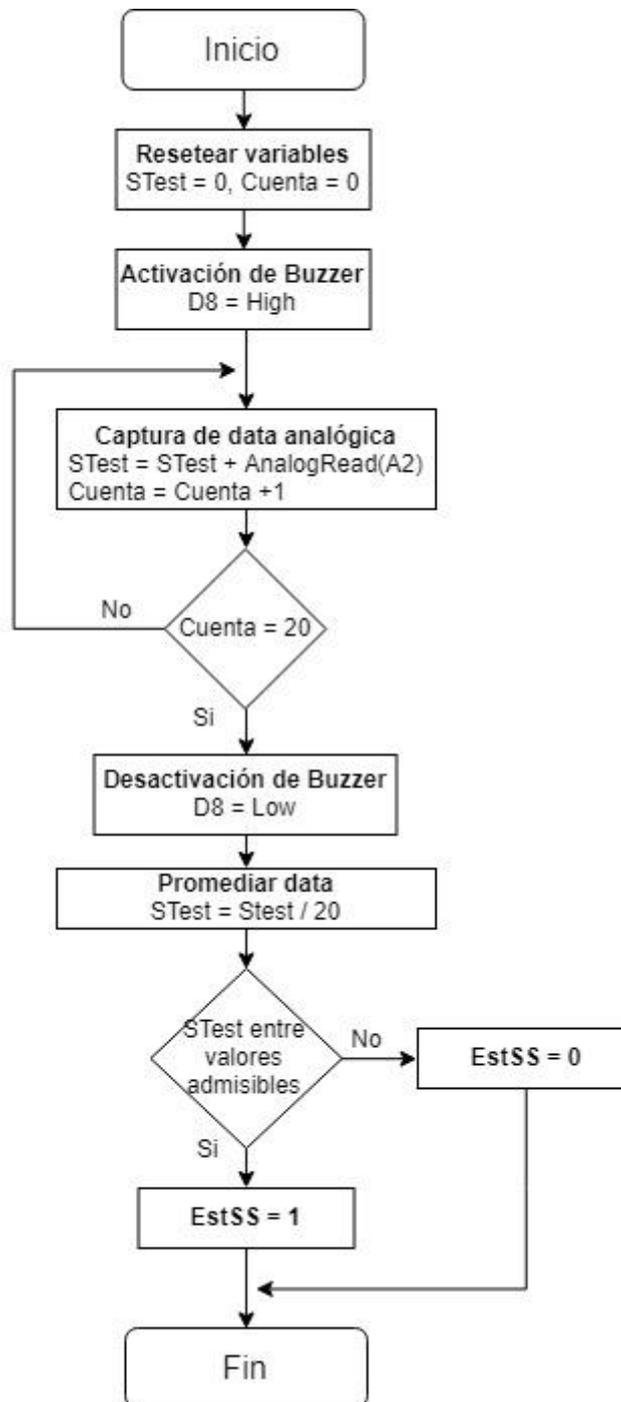


Figura 4.12 Diagrama de flujo de Verificación de estado de sensor

4.2.7.4. Verificar alimentación del nodo sensor

Esta función revisará si el nodo sensor está siendo alimentado la fuente primaria o está usando la batería. En la figura 4.11 se muestra este proceso.

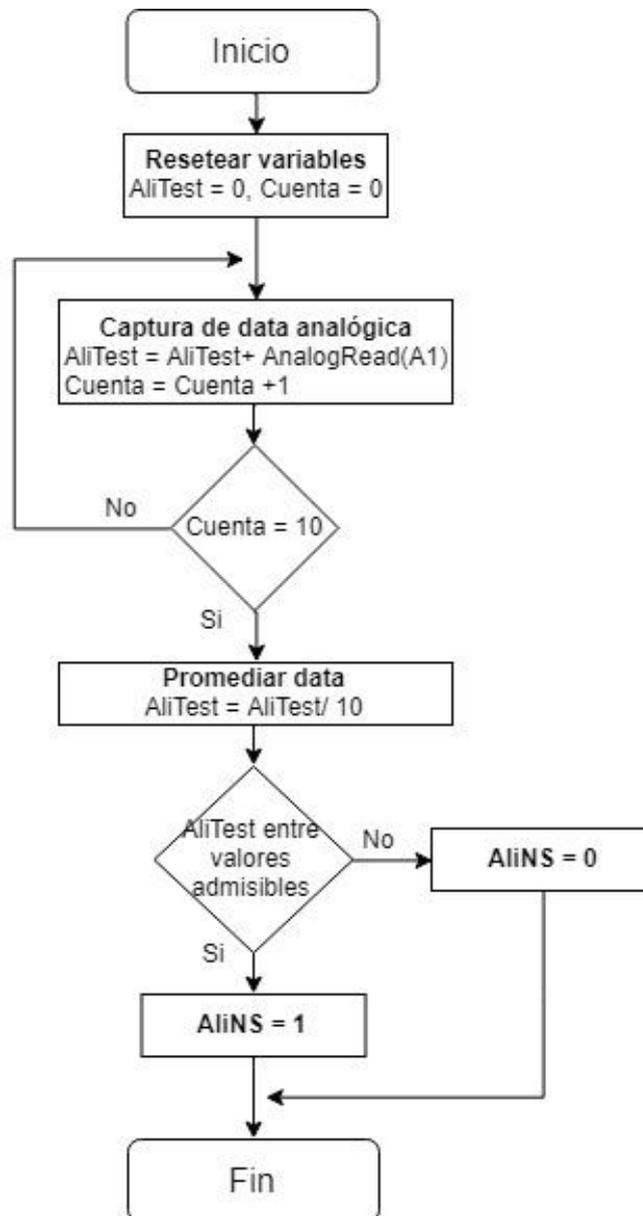


Figura 4.13 Diagrama de flujo de Verificación de alimentación NS

4.2.7.5. Revisar validez de data

Esta función utilizará la variable global CuentaVal para revisar si existen niveles de ruido suficientemente altos y ausencia de lluvia como para iniciar una medición para enviar al sistema Ciber. Esto le permitirá al nodo sensor ahorrar consumo energético y computacional ya que de no presentarse altos niveles o haber lluvia no se realizará la medición. En la figura 4.12 se muestra este proceso.

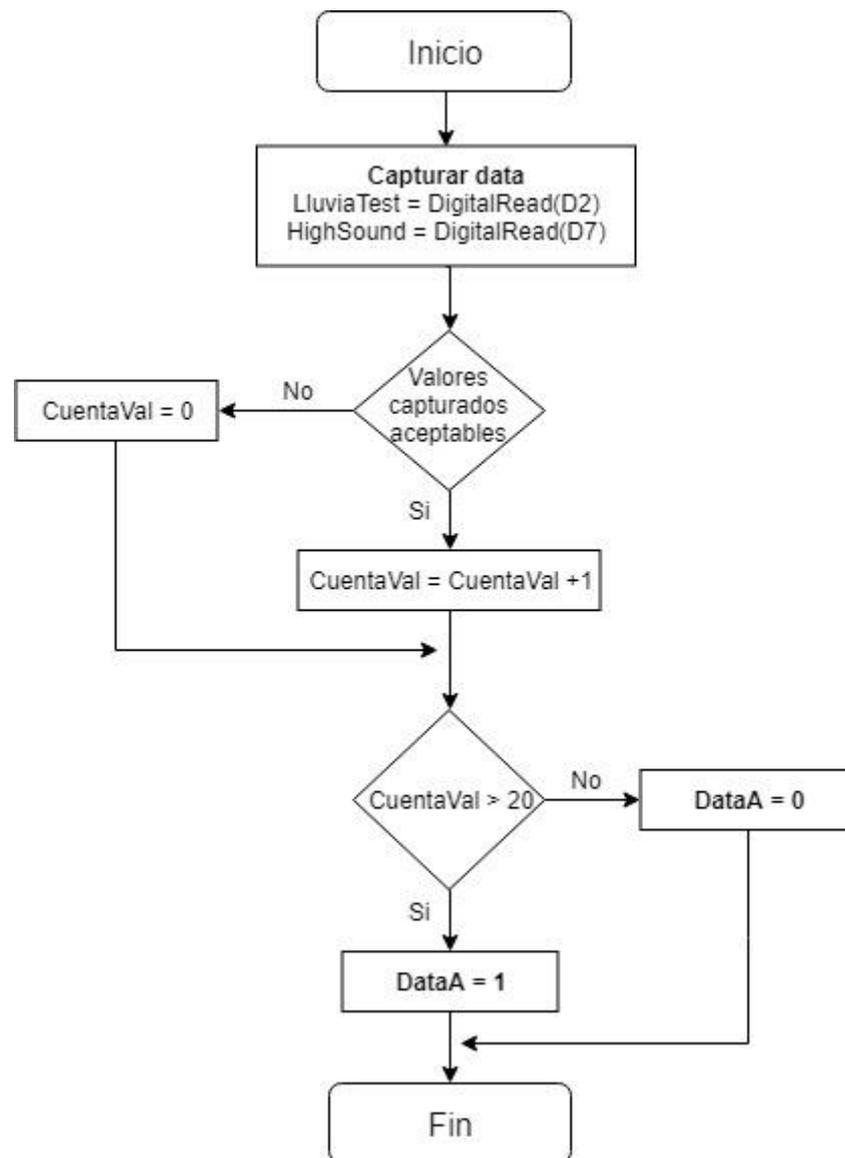


Figura 4.14 Diagrama de flujo de Validez de data

4.2.7.6. Medición y validación de data

Esta función capturará 60 datos de la presión sonora por un minuto, luego esta data será ponderada utilizando la fórmula de la sección 4.1 de este documento para finalmente dar ese valor como salida para el envío al sistema ciber. En la figura 4.13 se muestra este proceso.

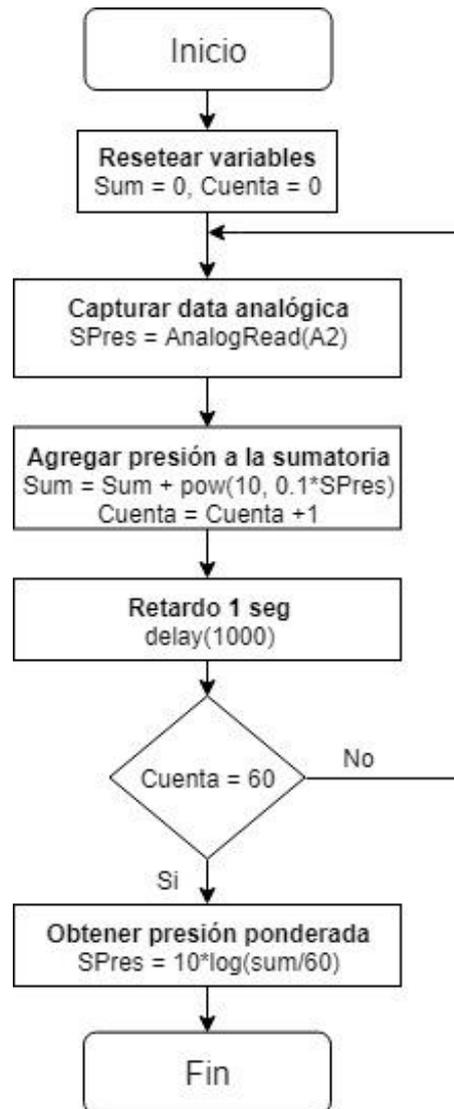


Figura 4.15 Diagrama de flujo de medición y validación

4.3. Diseño de arquitectura del sistema Ciber:

En esta sección se realizará el proceso de diseño de arquitectura de software ADD la cual fue explicada en la sección 2.5 de este documento.

4.3.1. Procesos de diseño ADD 3.0

4.3.1.1. Revisión de entradas (Paso 1)

El primer paso en este proceso de diseño nos pide tener claro el concepto general del sistema a desarrollar, para esto se hará una revisión detallada de cada una de las entradas previamente definidas.

Propósito de diseño

Definir una arquitectura basada en referencias para el sistema Ciber capaz de soportar la data proporcionada por el sistema físico considerando aspectos funcionales, comerciales, de seguridad, de tiempo de respuesta y de calidad de datos.

Requerimientos funcionales

Los requerimientos funcionales describen las funciones que debe lograr el sistema en su totalidad, estas características son vistas generalmente del punto de vista del cliente o usuario y deben demostrar el logro de algún objetivo en específico.

Los principales requerimientos funcionales del sistema ciber de este proyecto son descritos en la tabla 4.25:

Tabla 4.25 Descripción de objetivos de diseño

ID	Requerimientos	Descripción
UC-1	Recolectar data sensada	El sistema debe ser capaz de recibir, almacenar y procesar tramas de datos de los sensores identificando la información entregada (niveles de presión sonora, presencia de lluvia)
UC-2	Monitorear estado de sensores	El sistema debe realizar seguimiento diario del estado sus sensores.
UC-3	Actualizar información en tiempo real	El usuario debe recibir información del nivel de presión sonora de su zona, con data actualizada en tiempo real cada minuto
UC-4	Almacenar datos	El sistema debe almacenar toda la data registrada de forma ordenada
UC-5	Soportar análisis de datos	Se requiere una plataforma que pueda analizar la data recolectada por los sensores e identificar estadísticas de contaminación
UC-6	Monitorear estado de los subsistemas Ciber	El administrador deber poder ver el estado y funcionamiento de las distintas etapas que tendrá la arquitectura Cloud del sistema Ciber

Atributos de calidad

Los atributos de calidad son requerimientos no funcionales, estos generalmente cumplen objetivos de rendimiento, disponibilidad, modificabilidad, testeabilidad y seguridad. Estos requerimientos deben poseer características medibles las cuales deben ser descritas lo más específicas posibles de modo que se pueda confirmar su cumplimiento.

Para cumplir este objetivo, los atributos de calidad son descritos en formato de escenarios. La tabla 4.26 describe los escenarios de atributos de calidad del sistema ciber del presente proyecto.

Tabla 4.26 Escenarios de atributos de Calidad

ID	Atributo de calidad	Escenario
QA-1	Rendimiento	Se estima que hasta 8 nodos sensores podrán enviar data de manera simultánea a los Gateway. Con una etapa inicial de implementación de 10 Gateway el sistema deberá ser capaz de recibir hasta 80 datos en forma simultánea. Esta data debe ser procesada y almacenada correctamente en un tiempo no mayor a 3 segundos.
QA-2	Rendimiento	La interface visual deberá actualizarse automáticamente con una latencia no mayor a 5 segundos.
QA-3	Modificabilidad	El sistema debe ser capaz incluir nuevos nodos sensores y Gateway sin necesidad de cambios complejos en su estructura.
QA-4	Disponibilidad	La data de cada sensor debe seguir siendo transmitida aunque falle un Gateway
QA-5	Testeabilidad	El sistema debe poder ser evaluado en términos de velocidad de respuesta y soporte de información.
QA-6	Seguridad	Se debe tener un registro log de todos los eventos de operación del sistema.
QA-7	Seguridad	El acceso como administrador debe ser autorizado a través de autenticación de un usuario y contraseña.

Restricciones

Las restricciones son las limitaciones técnicas u organizacionales que posee un proyecto debido a, por ejemplo, tecnologías de implementación previamente determinadas por clientes o disponibilidad del mercado.

La tabla 4.27 presenta las restricciones asociadas al proyecto realizado

Tabla 4.27 Restricciones del proyecto

ID	Restricción
CON-1	La transmisión de datos de los nodos al servidor debe ser por el Sistema LoraWAN
CON-2	Se debe utilizar el servidor TTN (The things Network) para la conexión de los sensores con el sistema ciber
CON-3	Utilizar tecnología Cloud Computing para captura y procesamiento de datos.
CON-4	El acceso del administrador deber ser por una plataforma web
CON-5	El acceso de usuarios debe ser a través de una aplicación móvil
CON-6	Los Gateways deben tener una conexión permanente a internet
CON-7	El sistema ciber debe trabajar con la parte física ya diseñada
CON-8	Utilizar una arquitectura de referencia IoT para el diseño.

Preocupaciones arquitecturales

Las preocupaciones arquitecturales son decisiones de diseño que deben tomarse en cuenta, aunque no estén mencionadas de forma explícita en los requerimientos funcionales del sistema.

La tabla 4.28 presenta las preocupaciones asociadas al proyecto realizado

Tabla 4.28 Preocupaciones del proyecto

ID	Preocupación
CRN-1	Aprovechar el uso de tecnologías Open-source para minimizar los gastos de implementación de sistema
CRN-2	La interface móvil debe ser implementable en dispositivos Apple, Android y Windows
CRN-3	Utilizar interfaces de programación que faciliten el entendimiento de los códigos.
CRN-4	Las bases de datos utilizadas deberán tener un respaldo que evite perdida de información en caso de fallas
CRN-5	Utilizar protocolos de comunicación de bajo consumo de red

4.3.1.2. Iteración 1: Selección de arquitectura (Pasos 2 al 7)

Paso 2: Establecer objetivo de la iteración y seleccionar las entradas a ser consideradas en la iteración

El sistema a diseñar tiene diversos requerimientos que pueden ser solucionados en una o más iteraciones, el paso 2 del ADD inicia definiendo el objetivo de la iteración a realizar.

El objetivo de esta primera iteración es definir la arquitectura en la que se basará el sistema. Para esto se tomarán en cuenta las siguientes entradas del paso 1:

- UC-1 Recolectar data sensada
- UC-4 Almacenar Datos
- UC-5 Soportar análisis de datos
- QA-3 Modificabilidad
- QA-6 Seguridad
- QA-7 Seguridad
- CON-1 La transmisión de datos de los nodos al servidor debe ser por el Sistema LoraWAN
- CON-3 Utilizar tecnología Cloud Computing para captura y procesamiento de datos.
- CON-4 El acceso administrador deber ser por una plataforma web
- CON-5 El acceso de usuarios debe ser a través de una aplicación móvil
- CON-7 El sistema ciber debe trabajar con la parte física ya diseñada.
- CON-8 Utilizar una arquitectura de referencia IoT para el diseño.
- CRN-5 Utilizar protocolos de comunicación de bajo consumo de red.

Paso 3: Elegir uno o más elementos del sistema para descomponer

Ya que se está definiendo la arquitectura a utilizar, la descomposición será del sistema en conjunto

Paso 4: Escoger uno o más conceptos de diseño para satisfacer las entradas seleccionas para la iteración

En la tabla 4.29 se muestra la arquitectura de referencia IoT (CRN-4) seleccionada.

Tabla 4.29 Decisiones de diseño – Paso 4

Decisiones de diseño y alojamiento	Rationale	
Se selecciona la arquitectura de referencia “Microsoft Azure IoT”	<p>Microsoft Azure IoT es una arquitectura serverless basada en microservicios y cloud computing (CON-3, CON-8) compuesta por 3 capas definidas como “Things”, “Insight” y “Actions”. Cada capa contiene subsistemas deben ser implementados como un servicio independiente y con escalabilidad independiente (QA-3).</p> <p>Principales Subsistemas:</p> <ol style="list-style-type: none"> 1. IoT Devices: Tiene la capacidad de recolectar data del exterior para posteriormente enviarla a la nube (UC-1). 2. Cloud Gateway: Recibe y procesa la data los IoT Devices de forma segura. 3. Stream Processing: Consume la data del Gateway, la integra con el Bussines Processes y la coloca en el Storage (UC-4, UC-5) 4. User Interface: Permite visualizar la data recolectada y monitorear los dispositivos IoT <p>Subsistemas opcionales:</p> <ol style="list-style-type: none"> 1. IoT Edge Devices: Además de las características básicas de los IoT Devices, estos dispositivos poseen la capacidad de procesamiento local para enviar una data más trabajada a la nube. 2. Data transformation: Se encarga de reestructurar, combinar o transformar la data enviada por los dispositivos IoT. 3. Machine Learning: Permite la ejecución de algoritmos predictivos sobre la data recolectada para mejorar su procesamiento. 4. User Management: Permite dividir las funcionalidades del sistema entre distintos tipos de usuarios y controlar los accesos. 	
	Alternativa	Motivo de descarte
	RAMI 4.0	Arquitectura IoT orientada a ambientes industriales con alta variedad de data y procesos, además de distancias cortas de transmisión de datos.
	IoT-A	Arquitectura bastante detallada orientada a proyectos de alta complejidad como SmartCity lo cual no forma parte del alcance de esta tesis

En la figura 4.16 se muestra la vista de capas de la arquitectura de referencia seleccionada y descrita en la tabla 4.29. Esta arquitectura de referencia usa como base el patrón de arquitectura Lambda.

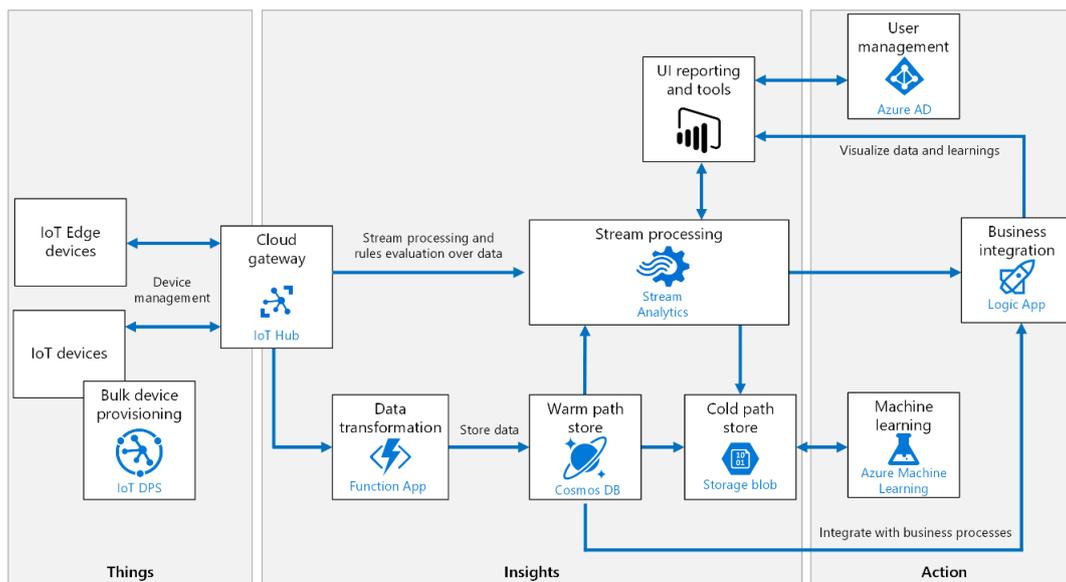


Figura 4.16 Arquitectura Microsoft Azure IoT

Fuente: [27]

Paso 5: Instanciar los elementos arquitecturales, asignar responsabilidades y definir interfaces

Tabla 4.30 Decisiones de diseño – Paso 5

Decisiones de diseño y alojamiento	Rationale
Eliminar el subsistema Machine Learning	En la arquitectura de referencia de Microsoft Azure, se utiliza el Machine Learning para tratamiento de la data proveniente del procesamiento Batch. Esto es útil cuando se trabaja con Big Data sin embargo como el sistema en desarrollo centra su trabajo únicamente en el monitoreo de la contaminación acústica, este subsistema es innecesario.
Agregar el subsistema “Physical Gateway”	Ya que el sistema trabajará con la transmisión de data de los sensores a través del sistema LoraWAN (CON-1), se debe cumplir su estructura de comunicación la cual requiere de un Gateway físico para la recolección de data de los sensores. Por esto al subsistema “Cloud Gateway” se le agregará como etapa previa “Physical Gateway”.
Eliminar el subsistema IoT Device	En la arquitectura de referencia de Microsoft Azure contiene 2 tipos de dispositivos IoT en su etapa inicial, sin embargo, el dispositivo a utilizar ya está definido (CON-7) y es del tipo Edge IoT ya que tiene un procesamiento local de la data. Debido a esto se eliminara el otro modelo de dispositivo.
Agregar capa “Output Devices”	Se agregó esta capa con el propósito de identificar los 2 tipos de interfaces de usuario que presentara el sistema. <ul style="list-style-type: none"> • Real Time Interface: Mostrará la información inmediata del nivel de presión sonora para cualquier usuario que ingrese a la aplicación (CON-5). • Batch Interface: Mostrará los registros de todas las mediciones tomadas por los sensores (CON-4).
Definir MQTT como protocolo de comunicación	El protocolo fue seleccionado por los siguientes motivos: <ul style="list-style-type: none"> ▪ Es un protocolo ligero que soporta data LoraWAN (CON-1) ▪ Bajo consumo de energía y red (CRN-5) ▪ Protocolo orientado a IoT

Paso 6: Boceto de las vistas y registro de las decisiones de diseño

En la figura 4.17 se muestra el boceto de la arquitectura teniendo en consideración las decisiones de diseño planteadas en los pasos previos.

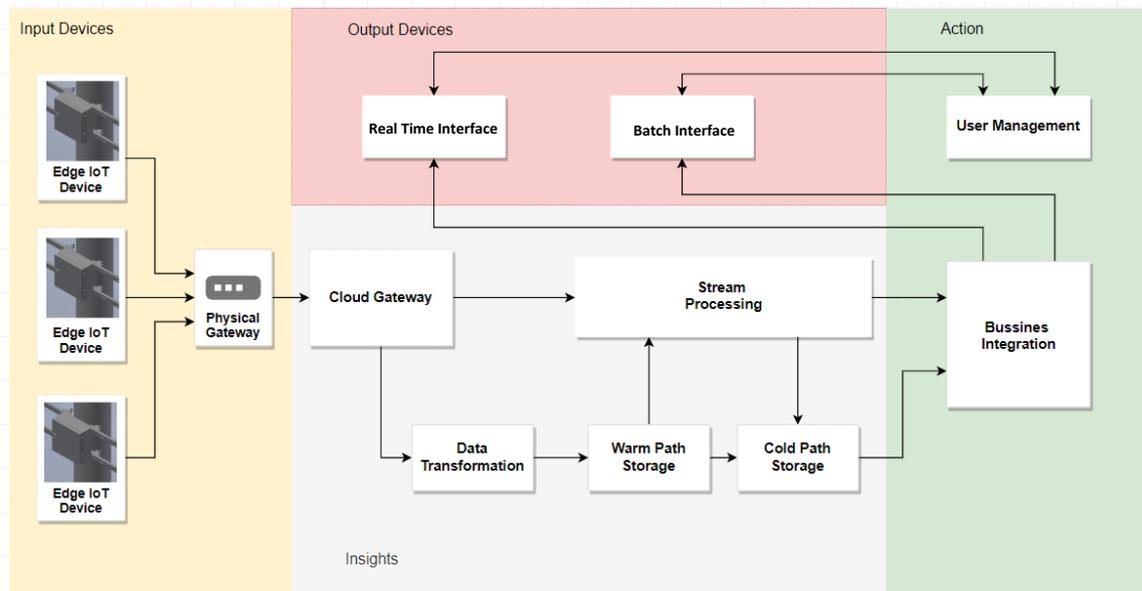


Figura 4.17 Arquitectura Inicial

La tabla 4.31 describe cada subsistema de la arquitectura diseñada:

Tabla 4.31 Descripción de subsistemas

Subsistema	Descripción
Physical Gateway	Se encarga de recolectar la data de los nodos sensores que se encuentren en su rango
Warm Path Storage	Este elemento almacena los datos obtenidos por un corto periodo predeterminado, y su principal función es tener la data lista para ser procesada o mostrada en fracciones de segundo.
Cold Path Storage	Este elemento almacena toda la data registrada por los sensores para futuros procesamientos.
UI & Reporting Tools	Muestra la data procesada y los reportes a través de interfaces visuales como sitios web o aplicaciones móviles.
Real Time Interface	Este elemento mostrará la información solicitada por el usuario a través de un aplicativo móvil o web.
Batch Interface	Mostrará los registros de toda la data recolectada para su análisis por un usuario administrador

Paso 7: Realizar el análisis del diseño actual, revisar el objetivo de la iteración y el logro del propósito del diseño

Para el análisis de esta iteración se utilizará la tabla 4.32 que sigue el diseño Kanban como se muestra a continuación.

Tabla 4.32 Revisión de objetivos

No Realizado	Parcialmente Realizado	Completamente Realizado	Decisiones de diseño realizadas durante la iteración
-	-	UC-1	Arquitectura basada en Azure IoT que posee subsistemas especializados en recolección, almacenamiento y procesamiento de datos a gran escala y rápida respuesta.
-	-	UC-4	
-	-	UC-5	
-	QA-3	-	Modelo arquitectónico con subsistemas independientes unos de otros, permitiendo una sencilla escalabilidad
QA-6	-	-	No se han tocado temas de monitoreo de sistema en esta iteración.
-	QA-7	-	Uso de subsistema “User Management” para autenticación de usuarios.
-	CON-1	-	División de Gateway en sistema físico y Cloud con el fin de seguir la estructura LoraWAN
-	CON-3	-	Arquitectura de referencia basada en sistemas de Cloud Computing
-	-	CON-4	Creación de capa “Output Devices” con el propósito de diferenciar las vistas de la interface móvil de la interface web del sistema
-	-	CON-5	
-	-	CON-7	Selección de subsistema de captura de registro de datos pensado en los nodos Sensores del sistema físico
-	-	CON-8	Selección de una arquitectura IoT como base de esta iteración
-	-	CRN-5	Selección de protocolo de comunicación MQTT por su bajo consumo de red

4.3.1.3. Iteración 2: Flujo de datos y monitoreo (Pasos 2 al 7)

Paso 2: Establecer objetivo de la iteración y seleccionar las entradas a ser consideradas en la iteración

Ahora que se tiene una arquitectura definida, el objetivo de esta segunda iteración es determinar la estructura del flujo de datos del sistema:

- UC-2 Monitorear estado de sensores
- UC-3 Actualizar información en tiempo real
- UC-6 Monitorear estado de los subsistemas Ciber
- QA-2 Rendimiento
- QA-6 Seguridad

Paso 3: Elegir uno o más elementos del sistema para descomponer

Se descompondrán principalmente los elementos del subsistema Stream Processing y algunos sistemas dependientes de este.

Paso 4: Escoger uno o más conceptos de diseño para satisfacer las entradas seleccionadas para la iteración

Tabla 4.33 Decisiones de diseño – Paso 4

Decisiones de diseño y alojamiento	Rationale
Se selecciona la arquitectura de referencia Stateless Stream Processing	<p>La arquitectura stateless provee una solución rápida y escalable para análisis de datos en tiempo real (UC-3), este tipo de arquitectura reduce el costo operacional y aumenta la disponibilidad del sistema sin embargo solo puede implementarse cuando la información de sus estados no requiere ser mantenida durante el procesamiento.</p> <p>Características:</p> <ul style="list-style-type: none"> • Permite identificar y reaccionar ante eventos específicos cuando no se requiere tomar en consideración la data histórica. Por ejemplo, fallas en los sensores. (UC-2) • Permite transmitir directamente toda la telemetría proveniente de los sensores para su almacenamiento volátil y/o visualización (QA-2). • Adicionalmente toda la data procesada puede ser almacenada permanentemente en una base de datos para un futuro procesamiento Batch

	Alternativa	Motivo de descarte
	Stateful Stream Processing	Supera las limitaciones de la arquitectura Stateless a consta de un mayor costo operacional. Sin embargo, el sistema en cuestión no presenta dicha necesidad.
Añadir servicio de monitoreo Cloud	Con el fin de satisfacer los requerimientos UC-6 y QA-8, se adicionara un sistema de monitoreo Cloud que permita al cliente tener un mejor control sobre todos los subprocesos del sistema	

En la figura 4.18 se muestra la arquitectura de referencia seleccionada y descrita en la tabla 4.33. Esta arquitectura de referencia usa como base el patrón de arquitectura Lambda.

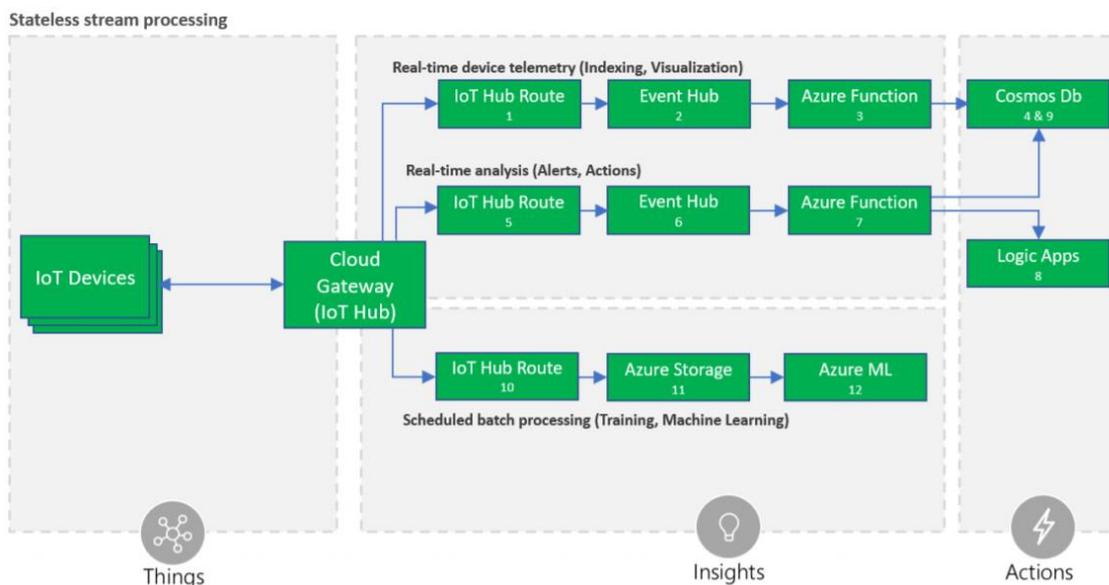


Figura 4.18 Modelo de arquitectura Stateless basada en Azure
Fuente: [27]

Paso 5: Instanciar los elementos arquitecturales, asignar responsabilidades y definir interfaces

Tabla 4.34 Decisiones de diseño – Paso 5

Decisiones de diseño y alojamiento	Rationale
Dividir capa Insight en una capa Real	La división de esta capa permitirá identificar qué tipo de datos recibirán un procesamiento en tiempo real (UC-3) para una rápida visualización y

Time y una capa Batch	cuáles serán almacenados en y procesados para análisis más especializados.
Division de subproceso “Stream Processing”	Existirán 3 rutas en el procesamiento del flujo de datos, una para la visualización en la aplicación móvil, una para las alertas de estados de los sensores (UC-2) y una para el almacenamiento batch.
División de subprocesos “Bussines Integration”	Se dividirá el subprocesos “Bussines Integration” para tener un procesamiento independiente entre lo que solicitara la interfase móvil con lo que solicitara la interfase web.
Adicionar capa de monitoreo	Esta capa utilizara un servicio de monitoreo Cloud existente en el mercado y se encargara de controlar e informar el estado y las acciones de todas las etapas de la arquitectura del sistema (UC-6, QA-6)

Paso 6: Boceto de las vistas y registro de las decisiones de diseño

En la figura 4.19 se muestra el boceto de la arquitectura teniendo en consideración las decisiones de diseño planteadas en los pasos previos.

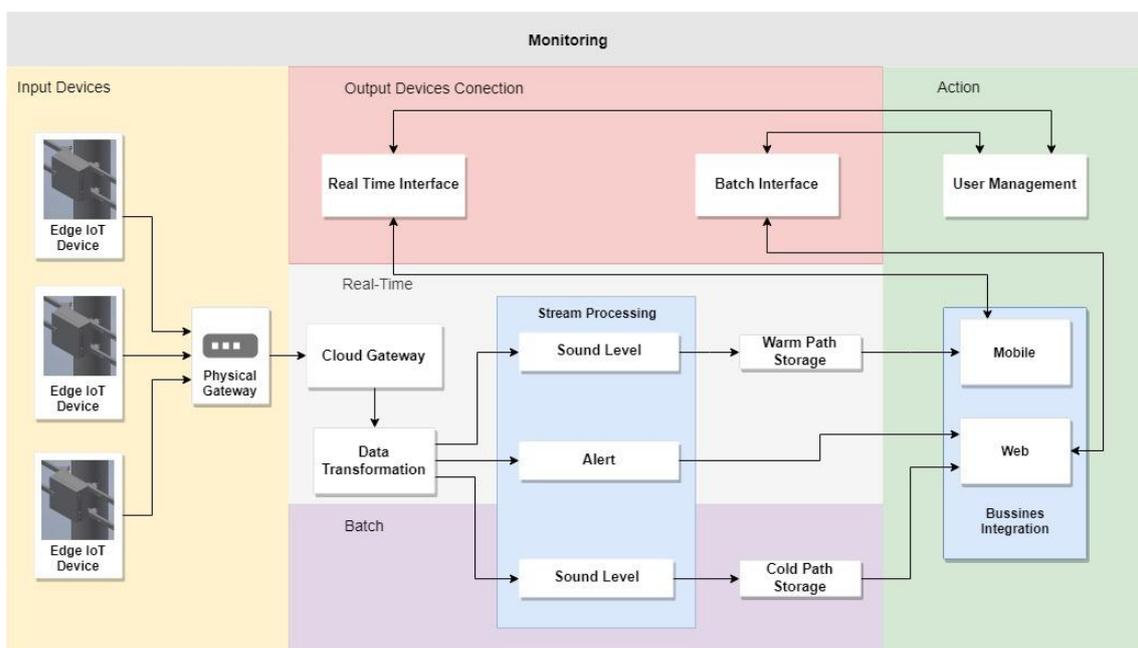


Figura 4.19 Arquitectura Iteración 2

Paso 7: Realizar el análisis del diseño actual, revisar el objetivo de la iteración y el logro del propósito del diseño

Para el análisis de esta iteración se utilizará la tabla 4.35 que sigue el diseño Kanban como se muestra a continuación.

Tabla 4.35 Revisión de objetivos

No Realizado	Parcialmente Realizado	Completamente Realizado	Decisiones de diseño realizadas durante la iteración
-	-	UC-2	División de subproceso “Stream Processing” para establecer una ruta encargada de la revisión del estado de los sensores en tiempo real.
-	-	UC-3	División de subproceso “Stream Processing” para establecer una ruta encargada de transmitir el nivel de presión en tiempo real a la interfase movil.
-	-	UC-6	Adicionamiento de una capa de monitoreo para asegurar el correcto funcionamiento del sistema Cloud
-	QA-2	-	Creación de capa real-time enfocada en transmisión inmediata de la data recolectada evitando problemas de latencia.
-	QA-6	-	Posibilidad de obtener registro log de todos los eventos de operación del sistema a través del subsistema de la capa de monitoreo.

4.3.1.4. Iteración 3: Selección de Tecnologías (Pasos 2 al 7)

Paso 2: Establecer objetivo de la iteración y seleccionar las entradas a ser consideradas en la iteración

Ahora que se tiene una arquitectura definida, el objetivo de esta segunda iteración es la selección de tecnologías a utilizar, para esto se tomarán en consideración las siguientes entradas:

- QA-1 Rendimiento
- QA-2 Rendimiento
- QA-3 Modificabilidad
- QA-4 Disponibilidad
- QA-6 Seguridad
- QA-7 Seguridad
- CON-1 La transmisión de datos de los nodos al servidor debe ser por el Sistema LoraWAN.
- CON-2 Se debe utilizar el servidor TTN (The things Network) para la conexión de los sensores con el sistema ciber.

- CON-3 Utilizar tecnología Cloud Computing para el procesamiento de datos.
- CRN-1 Aprovechar el uso de tecnologías Open-source para minimizar los gastos de implementación de sistema

Paso 3: Elegir uno o más elementos del sistema para descomponer

Se descompondrán los elementos de la arquitectura definida para poder definir mejor las tecnologías a utilizar.

Paso 4: Escoger uno o más conceptos de diseño para satisfacer las entradas seleccionas para la iteración

Tabla 4.36 Decisiones de diseño – Paso 4

Decisiones de diseño y alojamiento	Rationale	
Se selecciona XaaS como modelo de servicio de Cloud Computing.	<p>El modelo XaaS abarca los 3 principales modelos de servicio de mercado Cloud (CON-3):</p> <ul style="list-style-type: none"> • Software as a Service (SaaS): El cliente usa las aplicaciones del proveedor en una infraestructura de nube, en este modelo el cliente no controla o administra la infraestructura, sino que únicamente la utiliza. • Platform as a Service (PaaS): El consumidor despliega sus propias aplicaciones en la infraestructura entregada por el proveedor, en este modelo el consumidor nuevamente no posee control de la infraestructura, pero puede realizar algunas variaciones ambientales como el incremento de espacio de almacenamiento. • Infrastructure as a Service (IaaS): El proveedor proporciona recursos computacionales como redes, procesamiento, almacenamiento y otros elementos básicos en donde el cliente puede ejecutar administrar y monitorear los sistemas operativos, el almacenamiento y las aplicaciones. 	
Se selecciona AWS como proveedor de tecnología Cloud	Según la revista Forbes los principales proveedores de servicios Cloud son Microsof, Amazon, IBM, Sales Force y SAP. Entre estos, Amazon actualmente posee una conectividad directa con TTN a través de AWS IoT lo cual permite una fácil conectividad de dispositivos LoRa con la nube de Amazon. (CON-2).	
	Alternativa	Motivo de descarte
	Mircrosoft Azure IoT	Compatibilidad con TTN aun en desarrollo lo cual limita el uso de recursos y dificulta la programación entre estos servidores

Paso 5: Instanciar los elementos arquitecturales, asignar responsabilidades y definir interfaces

Tabla 4.37 Decisiones de diseño – Paso 5

Decisiones de diseño y alojamiento	Rationale
<p>Seleccionar AWS IoT y TTN como tecnologías para el subsistema “Cloud Gateway”</p>	<p>The Things Network (TTN): Los sistemas backend de The Things Network son responsables de enrutar los datos IoT entre dispositivos y aplicaciones. Este servidor a, diferencia de las redes IoT convencionales, puede enrutar paquetes de datos que no utilizan protocolos IP como lo son los dispositivos LoRa lo cual lo hace indispensable en el desarrollo de aplicación que utilicen esta tecnología [28].</p> <p>AWS IoT: Gracias al servicio gratuito de Cloudformation que posee AWS, se pueden conectar directamente los dispositivos registrados en TTN a la nube de Amazon lo cual permite desarrollar aplicaciones de manera convencional y permite una comunicación bidireccional entre los dispositivos y la nube AWS a través de MQTT. [29] AWS IoT es una plataforma PaaS que se componen de los siguientes elementos:</p> <p>IoT Policy: “Cada dispositivo conectado debe tener credenciales para obtener acceso al agente de mensajes o al servicio Device Shadow. Todo el tráfico hacia o desde AWS IoT debe estar cifrado sobre el protocolo Transport Layer Security (TLS). Las credenciales de dispositivos deben guardarse en un lugar seguro para que los datos se envíen de forma segura al agente de mensajes” [29].</p> <p>IoT Topic: “es el agente de mensajes de AWS IoT es un servicio de agente de publicación/suscripción que permite enviar y recibir mensajes de AWS IoT. Durante la comunicación con AWS IoT, un cliente envía un mensaje dirigido a un Topic” [29].</p> <p>IoT Rule: “Las reglas permiten a sus dispositivos interactuar con los servicios de AWS. Las reglas se analizan y las acciones se ejecutan en función del flujo de temas MQTT. Puede utilizar reglas para admitir tareas como guardar archivos en DynamoDB, capturar una métrica en CloudWatch, enviar datos de mensajes a una entrada de eventos de AWS IoT, etc” [29].</p>
<p>Seleccionar DynamoDB como tecnología de almacenamiento para “Warm Path Storage”</p>	<p>“Amazon DynamoDB es una base de datos de claves-valor y documentos que ofrece rendimiento en milisegundos de un solo dígito a cualquier escala. Se trata de una base de datos duradera de varias regiones y con varios maestros, completamente administrada, que cuenta con copia de seguridad, restauración y seguridad integradas, y almacenamiento de caché en memoria para aplicaciones a escala de Internet. DynamoDB puede gestionar más de 10 billones de solicitudes por día y puede admitir picos de más de 20 millones de solicitudes por segundo” [30].</p>

<p>Seleccionar Amazon S3 como tecnología de almacenamiento para “Cold Path Storage”</p>	<p>“Amazon Simple Storage Service (Amazon S3) es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos, seguridad y rendimiento. Amazon S3 está diseñado para ofrecer una durabilidad del 99,999999999 % y almacena datos de millones de aplicaciones para empresas de todo el mundo” [31].</p>
<p>Seleccionar AWS Lambda como tecnología para el procesamiento de data para la plataforma web y móvil.</p>	<p>“AWS Lambda es un servicio informático que permite ejecutar código sin aprovisionar ni administrar servidores. AWS Lambda ejecuta el código solo cuando se le solicita, y se escala de manera automática, lo cual le permite pasar de pocas solicitudes al día a miles por segundo” [32].</p>
<p>Seleccionar Api Gateway como tecnología para generar acceso a consultas de data en tiempo real.</p>	<p>“Amazon API Gateway es un servicio que facilita a los desarrolladores la creación, la publicación, el mantenimiento, el monitoreo y la protección de API a cualquier escala. Este servicio permite crear API REST y API WebSocket que actúen como "puerta delantera" para que las aplicaciones obtengan acceso a datos, lógica de negocio o funcionalidades desde servicios backend como AWS Lambda” [33].</p>
<p>Seleccionar Amazon CloudFront como tecnología para distribuir el contenido batch.</p>	<p>“Amazon CloudFront es un servicio rápido de red de entrega de contenido que distribuye datos, vídeos, aplicaciones y API de forma segura, con baja latencia, altas velocidades de transferencia y dentro de un entorno fácil para desarrolladores” [34].</p>
<p>Seleccionar Amazon Cognito como tecnología para la autenticación de usuarios.</p>	<p>“Amazon Cognito permite incorporar a aplicaciones web y dispositivos móviles funcionalidades como el control de acceso, la inscripción y el inicio de sesión de los usuarios. El escalado de Amazon Cognito permite admitir millones de usuarios e iniciar sesión tanto mediante proveedores de identidad social e identidad empresarial” [35].</p>
<p>Seleccionar Amazon Cloud Watch como tecnología para el monitoreo de los procesos en la nube</p>	<p>“Amazon CloudWatch es un servicio de monitoreo y administración cloud. CloudWatch ofrece datos e información procesable para monitorear aplicaciones, comprender cambios de rendimiento que afectan a todo el sistema y tomar acciones, optimizar el uso de recursos y lograr una vista unificada del estado de las operaciones” [36].</p>

Paso 6: Boceto de las vistas y registro de las decisiones de diseño

En la figura 4.20 se muestra el boceto de la arquitectura teniendo en consideración las decisiones de diseño planteadas en los pasos previos.

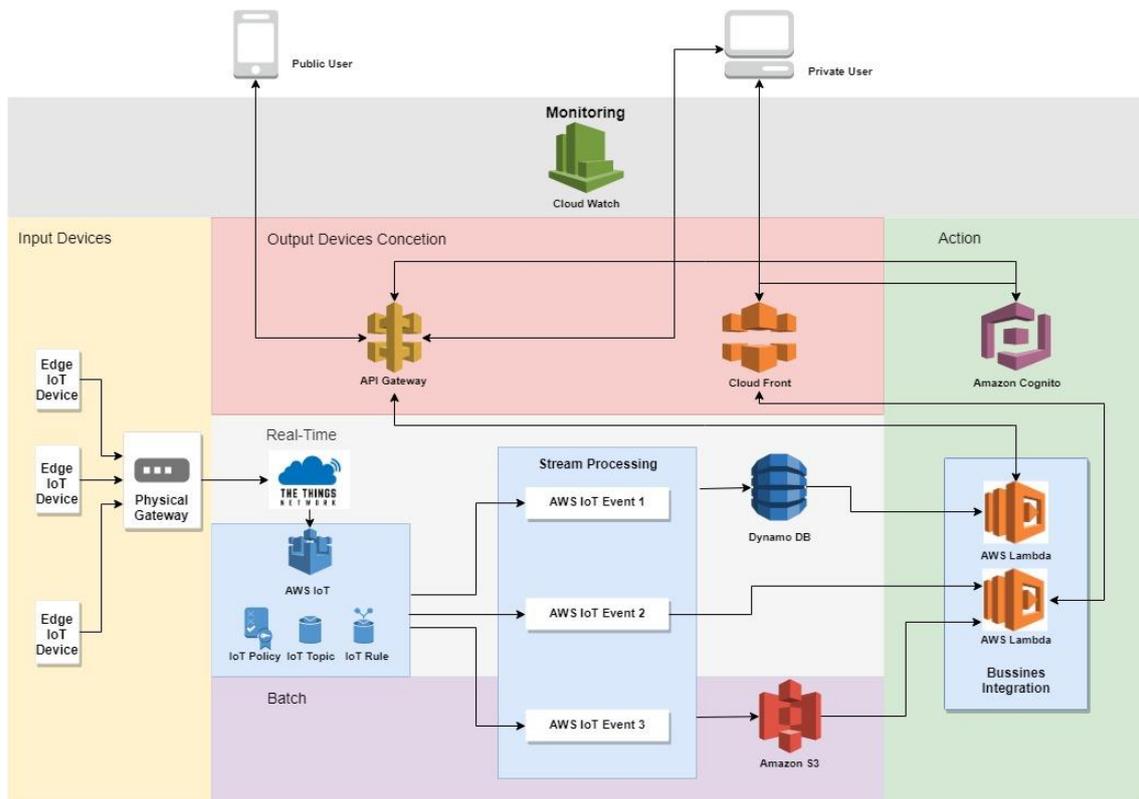


Figura 4.20 Arquitectura Iteración 2

Paso 7: Realizar el análisis del diseño actual, revisar el objetivo de la iteración y el logro del propósito del diseño

Para el análisis de esta iteración se utilizará la tabla 4.38 que sigue el diseño Kanban como se muestra a continuación.

Tabla 4.38 Revisión de objetivos

No Realizado	Parcialmente Realizado	Completamente Realizado	Decisiones de diseño realizadas durante la iteración
-	-	QA-1	Uso de base de datos NoSql DynamoDB con capacidad de lectura y escritura en tiempo real que puede superar fácilmente las 200000 lecturas por segundo.
-	-	QA-2, CON-3	Procesamiento de data móvil realizada por AWS Lambda, un servicio Cloud que comienza con un máximo de 500 ejecuciones en simultáneo y se escala cada minuto en función a las necesidades del sistema.

-	-	QA-3, CON-1	Uso del servidor TTN para la conexión de dispositivos físicos LoraWan, el cual permite un adiconamiento de Gateways y nodos sensores de forma sencilla y sin límites.
-	-	QA-4	Uso de topología Estrella-Estrella por parte del TTN lo cual permite seguir recibiendo data de los nodos sensores siempre que tenga al menos un Gateway en su rango.
-	-	QA-6	Uso de servicio Amazon Cloudwatch para el monitoreo del sistema Cloud proporcionando Logs de todos los eventos que analiza.
-	-	QA-7	Uso de servicio Amazon Cognito para la autenticación de usuarios a la plataforma web
-	-	CON-1	Conexión de TTN con la nube de Amazon a través servicio Cloudformation de AWS.
CRN-1	-	-	La facilidad de implementación, compatibilidad, respaldo y documentación brindada por AWS justifica el costo de sus servicios.

5. CAPÍTULO V: PROTOTIPO Y PRUEBAS DEL SISTEMA

Para poder realizar pruebas de alcance y funcionamiento del sistema se realizarán prototipos del Gateway y del Nodo Sensor. El Gateway estará compuesto por un Raspberry Pi 3 Model B+ y en lugar del RAK2245 se utilizará un transceptor LoRa simple (el mismo utilizado en el nodo sensor), este Gateway será de 1 solo canal lo cual limita bastante su capacidad de recibir información de múltiples nodos de manera simultánea, pero cumple su función como dispositivo de prueba. El nodo sensor solo contará con el sensor de sonido para la prueba del mismo, el transceptor LoRa y un pulsador que servirá para las pruebas de alcance de la señal por radiofrecuencia.

5.1. Prototipo Gateway

Para la construcción de este prototipo se utilizará de referencia la guía virtual elaborada por Chris Samuelson [37]. Esta guía enseña el proceso de conexión y programación de un Gateway de 1 solo canal, en la figura 5.1 se puede apreciar el prototipo armado; este cuenta con una Raspberry Pi 3 model B+, un transceptor LoRa modelo RFM95W y una antena de 915 MHz.



Figura 5.1 Prototipo Gateway

Una vez armada la parte física se pasó a programar la Raspberry, para ello se utilizó el código de la guía de referencia y se editó para cumplir con las características requeridas por el sistema. En la figura 5.2 se aprecia el programa principal al cual se le configuraron los pines de la Raspberry, ubicación, el servidor web y la frecuencia de transmisión y recepción; para el servidor se utilizará el de Estados Unidos y la frecuencia será de 905.5 MHz la cual es aceptada para ser utilizada en el Perú según la documentación del servidor TTN.

```

// SX1272 - Raspberry connections
int ssPin = 6;
int dio0 = 7;
int RST = 0;

// Set spreading factor (SF7 - SF12)
sf_t sf = SF7;

// Set center frequency
uint32_t freq = 905500000; // in Mhz! (905.5)

// Set location
float lat=12.15001112;
float lon=76.98916870;
int alt=60;

/* Informal status fields */
static char platform[24] = "Single Channel Gateway"; /* platform definition */
static char email[40] = ""; /* used for contact email */
static char description[64] = ""; /* used for free form description */

// define servers
// TODO: use host names and dns
#define SERVER1 "13.66.213.36" // The Things Network: croft.thethings.girovito.nl
#define PORT 1700 // The port on which to send data

```

Figura 5.2 Programa principal Gateway

Una vez configurado el Gateway se pasará a registrarlo en TTN, para esto se colocará el ID del Gateway el cual es mostrado por la consola de la Raspberry al correr el programa; además se colocarán los mismos valores de ubicación, servidor y frecuencia que se pusieron en el programa principal. En la figura 6.3 se muestra el Gateway registrado y conectado a la red TTN de forma satisfactoria.

The screenshot shows the 'GATEWAY OVERVIEW' page in the TTN interface. It displays the following information:

- Gateway ID:** eui-b827ebfffd0fc1d
- Description:** test Gateway
- Owner:** renzomurata (with a profile icon) and a link to 'Transfer ownership'.
- Status:** connected (indicated by a green dot).
- Frequency Plan:** United States 915MHz
- Router:** ttn-router-us-west
- Gateway Key:** A field containing a base64-encoded key, with a 'base64' label and a copy icon.
- Last Seen:** 40 minutes ago
- Received Messages:** 125
- Transmitted Messages:** 0

Figura 6.3 Registro de Gateway en TTN

5.2. Prototipo Nodo Sensor

Para la construcción de este prototipo se utilizará una placa Arduino UNO la cual posee las mismas características que el NANO, pero es de mayor tamaño y tiene una distribución de componentes que facilita su montaje en diseños de prueba como es nuestro caso. La conexión entre el Arduino y el transceptor LoRa será a través de los pines SPI estándar del Arduino, para el pulsador se utilizará el pin digital 7 y para el sensor de sonido el pin analógico 1. En la figura 5.4 se aprecia el prototipo del nodo sensor armado.

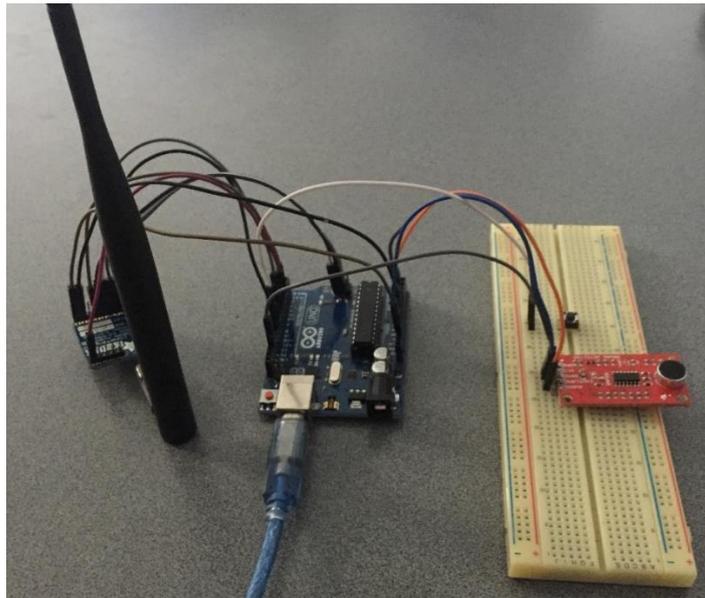


Figura 5.4 Prototipo Nodo Sensor

Antes de empezar la programación del Arduino se debe registrar la aplicación que conectará los nodos a la red TTN, se le definirá un ID y un servidor el cual deberá ser el mismo que de nuestro Gateway. Una vez creada se agregará un dispositivo al cual se le definirá un ID y un EUI el cual es un código de 8 bytes que identificará el dispositivo. En el método de activación se seleccionará ABP (Activation By Personalisation), con este método la data enviada por los sensores contendrá claves de identificación para poder ser recibidas por la red de forma segura. En la figura 5.5 se muestra el dispositivo “sensor1” registrado en la aplicación “renzo_lora_test”, en esta imagen se ven también los 3 códigos que serán necesarios programar en el Arduino para enviar la data a través de ABP estos son: “Device Address”, “Network Session Key” y “APP Session Key”.

DEVICE OVERVIEW

Application ID renzo_lora_test

Device ID sensor1

Activation Method ABP

Device EUI <> ⇄ 90 B4 5C 05 54 B1 E6 5A 📄

Application EUI <> ⇄ 70 B3 D5 7E D0 02 20 7B 📄

Device Address <> ⇄ 26 02 11 CD 📄

Network Session Key <> ⇄ 👁 📄

App Session Key <> ⇄ 👁 📄

Status ● 7 days ago

Figura 5.5 Registro de sensor en aplicación

Una vez completado el registro y habiendo copiado los códigos para el ABP se pasará a programar el Arduino y para ello se hará uso de la librería “MCCI LoRaWAN LMIC”; esta librería es una adaptación para Arduino de la librería LMIC (LoRaWAN-MAC-in-C) de IBM la cual permite utilizar dispositivos LoRa y conectarlos a la red TTN. Para este prototipo se utilizará de base uno de los Sketch de ejemplo de esta librería, pero adaptado a las funciones de nuestro sistema. En la figura 5.6 se muestra el código que utiliza la comunicación por ABP, aquí se puede apreciar que es necesario colocar las 3 claves de acceso durante la configuración, esto permitirá que toda la información transmitida por el sensor sea recibida por la red y enviada a la aplicación creada para su uso.

```
// LoRaWAN NwkSKey, network session key
// This should be in big-endian (aka msb).
static const PROGMEM u1_t NWKSKEY[16] = { 0xCD, 0x26, 0x57, 0x3C, 0xF8, 0xB3, 0xD2,

// LoRaWAN AppSKey, application session key
// This should also be in big-endian (aka msb).
static const PROGMEM APPSKEY[16] = { 0xEB, 0xC8, 0x63, 0xA5, 0x59, 0xC5, 0x66,

// LoRaWAN end-device address (DevAddr)
// See http://thethingsnetwork.org/wiki/AddressSpace
// The library converts the address to network byte order as needed, so this should
static const u4_t DEVADDR = 0x260211CD ; // <-- Change this address for every node!
```

Figura 5.6 Código 1 Nodo Sensor

Luego de escribir las claves de acceso y otras configuraciones como pines y frecuencia de transmisión, se pasará a desarrollar la lógica de prueba. Esta lógica tendrá 3 propósitos, probar que se produzca la conexión del nodo sensor con el Gateway, la conexión del Gateway con la red TTN, y el correcto sensado del nivel de presión sonora por parte del sensor de sonido. En la figura 5.7 se muestra el código del programa principal del Arduino, en este código al presionar el pulsador conectado a la entrada analógica 1 se iniciará un ciclo de 60 mediciones de sonido en un minuto; al finalizar se promediarán los resultados y se enviara la data a la aplicación en la red TTN.

```
void loop() {

  if (digitalRead(7) == LOW) { // pulsador presionado
    Serial.println(F("Button Pressed"));
    int count = 0;
    float soundS = 0;
    while (count < 60){ // bucle de 60 mediciones de sonido
      soundS = soundS + analogRead(1)*factor;
      count = count + 1;
      delay (1000);
    }
    sound = soundS/60; // promedio de mediciones
    do_send(&sendjob,sound); // funcion de envio de la data a buffer
    delay(50);
  }

  os_runloop_once(); // funcion ciclo de transmision y recepcion de data
}
```

Figura 5.7 Código 2 Nodo Sensor

Finalmente, en la aplicación creada en TTN se deberá definir una función para transformar la data enviada en una que pueda ser comprendida por los usuarios ya que el servidor recibe la data codificada y la decodifica a un formato hexadecimal. En la figura 5.8 se muestra la función que se encargará de transformar la data recibida a niveles de presión sonora, esta función también define el nombre del campo al cual se le asignará en base de datos.

```
1 function Decoder(bytes, port) {
2   if(bytes.length == 2) {
3     var soundP = (bytes[0]<<8) | bytes[1];
4     return {
5       'Presion sonora': soundP/ 100,
6     };
7   } else {
8     return {
9       'error': 'payload unknown'
10    };
11  }
12 }
```

decoder has unsaved changes [undo changes](#)

Figura 5.8 Decodificador en TTN

5.3. Pruebas de conectividad del sistema

Con los prototipos armados se pasará a probar la conexión con la web, el Gateway y la correcta decodificación de la data; para esta primera prueba se simplificará el código para enviar el valor de 75.23 decibeles cada vez que se presione el pulsador en el prototipo del nodo sensor. En la figura 5.9 se aprecian los resultados en la aplicación creada en la red TTN, en ello se ve que la data hexadecimal recibida en el “Payload” es decodificada por el código implementado en la etapa anterior y se obtiene como resultado los 75.23 decibeles enviados desde el prototipo de nodo sensor.



The screenshot shows the 'APPLICATION DATA' interface with a table of received data. The table has columns for time, counter, port, and payload. The payload is decoded to show 'PresionSonora: 75.23'.

time	counter	port	payload	Decoded Data
13:17:22	2	1	payload: 1D 63	PresionSonora: 75.23
13:17:15	1	1	payload: 1D 63	PresionSonora: 75.23
13:17:08	0	1	retry payload: 1D 63	PresionSonora: 75.23

Figura 5.9 Data recibida por la aplicación en TTN

5.4. Pruebas del sensor de sonido

Ya habiendo comprobado el correcto funcionamiento de las conexiones del prototipo, se pasará a probar que tan precisas son las mediciones del sensor seleccionado en comparación a un sonómetro profesional. Para esto, se hará uso del sonómetro con calibración vigente de la marca Hangzhou modelo AWA5661 mostrado en la figura 5.10, este es un sonómetro clase 1 que cumple con la normativa internacional para sonómetros IEC61672 y es utilizado para mediciones de contaminación acústica industrial y urbana.



Figura 5.10 Sonómetro AWA5661

Para esta prueba primero se calibrará el sensor del prototipo aplicando un factor al código de Arduino que iguale su medición con la del sonómetro profesional. En la figura 5.11 se aprecian los materiales a utilizar en las pruebas donde además del prototipo y el sonómetro, se utilizará un amplificador que cumplirá el rol de fuente de ruido.



Figura 5.11 Materiales para las pruebas

Para la calibración se emitió un ruido constante a través del amplificador el cual se regulo hasta 80 decibeles medidos con el sonómetro profesional, se utilizarán únicamente medias altas para estas pruebas debido a que la relación de nivel de presión sonora no es lineal y solo se buscara calibrar el sensor para altos niveles los cuales son nuestro punto de interés. El sonido se emitió por 1 minuto y con ambos sensores a una misma distancia del amplificador antes de tomar las medidas. En la figura 5.12 se muestra la medida del sonómetro profesional y en la figura 5.13 se muestran los resultados captados por el pin analógico del Arduino provenientes del sensor de sonido.



Figura 5.12 Medición de sensor profesional

```

20:19:48.760 -> 29
20:19:49.230 -> 29
20:19:49.733 -> 27
20:19:50.230 -> 28
20:19:50.765 -> 29
20:19:51.257 -> 28
20:19:51.742 -> 28
20:19:52.230 -> 28
20:19:52.733 -> 29
20:19:53.261 -> 29
20:19:53.762 -> 29
20:19:54.246 -> 26
20:19:54.732 -> 28
20:19:55.248 -> 28
20:19:55.748 -> 27
20:19:56.264 -> 28
20:19:56.728 -> 28

```

Figura 5.13 Mediciones del sensor de sonido

Los resultados obtenidos fueron que para una medición de 80.5dB, el sensor captó 20 valores en intervalos de un segundo a los cuales se le aplicó la fórmula de ponderación A explicada en la sección 4.1 de este documento. El resultado ponderado fue de 28.03 dB el cual nos da un factor de 2.87 para igualar ambas medidas. Con este factor se volverán a hacer las pruebas para mediciones de 65, 70, 75 y 85 dB con la finalidad comprobar si ambas medidas obtienen valores similares, estos resultados se mostrarán en la tabla 5.1.

Tabla 5.1 Resultados de pruebas del sensor de sonido

# de medición	Medición 1 (65.4dB)	Medición 2 (70.7dB)	Medición 3 (75.4dB)	Medición 4 (85.1dB)
1	63.14	71.75	74.62	83.23
2	66.01	68.88	77.49	83.23
3	66.01	74.72	80.92	86.10
4	60.27	66.01	80.92	86.10
5	66.01	68.88	74.62	88.97
6	63.14	71.75	71.75	86.10
7	68.88	66.01	74.62	88.97
8	63.14	71.75	77.49	86.10
9	63.14	71.75	74.62	86.10
10	63.14	68.88	77.49	86.10
11	66.01	68.88	74.62	83.23
12	63.14	74.72	71.75	83.23
13	66.01	71.75	74.62	88.97
14	68.88	68.88	74.62	86.10
15	63.14	66.01	74.62	88.97
16	60.27	71.75	74.62	88.97
17	63.14	71.75	71.75	86.10
18	60.27	74.72	83.23	86.10
19	66.01	71.75	74.62	86.10
20	60.27	68.88	77.49	88.97
Ponderado A	64.78	71.25	77.06	86.83

Como se aprecia en la tabla, la exactitud de la medición es $\pm 2\text{dB}$. Estos resultados son esperados ya que se utiliza un sensor de bajo costo el cual no posee la precisión para captar toda la gama de decibeles, además el uso del factor de conversión genera que el error también se amplifique lo cual no permite medidas muy certeras. El error obtenido, sin embargo, cumple con las exigencias plateadas en sistema y si bien no logra la precisión de un sonómetro profesional, permite identificar cuando hay presencia de contaminación sonora.

6. CAPÍTULO VI: COSTOS DEL SISTEMA

En este capítulo se analizarán los costos tanto del sistema físico como del sistema ciber, para esto se obtendrán costos individuales tanto del nodo sensor como del Gateway y luego se escalará a una posible primera implementación.

6.1. Costo del sistema Físico

6.1.1. Componentes eléctricos y electrónicos

Nodo Sensor

En la tabla 6.1 se muestra el costo de los componentes eléctricos y electrónicos de 1 nodo sensor.

Tabla 6.1 Costo de componentes eléctricos y electrónicos Nodo Sensor

Dispositivo	Modelo	Cantidad	Costo (\$)
Sensor de sonido	SEN-12642	1	10.95
Sensor de lluvia	FC-37	1	1.99
Sensor de corriente	ACS712-05	1	1.06
Microcontrolador	Arduino Nano	1	6.67
Buzzer	Groove Buzzer	1	2.07
Transceptor LoRa	RFM95W-915S2	1	13.84
Antena 915 MHz	WRL-14876	1	7.95
UPS	DD05CVSB_5V	1	3.18
Protección batería	DD05CVSA	1	0.95
Fuente AC/DC	PSK-S6C-5-T	1	17.11
Batería	803048	1	1.60
Tarjeta PCB 65 x110 mm	ALLPCB	1	2.2
Tarjeta PCB 60 x36 mm	ALLPCB	1	1
Costo Total			70.57

Gateway

En la tabla 6.2 se muestra el costo de los componentes eléctricos y electrónicos de 1 Gateway.

Tabla 6.2 Costo de componentes eléctricos y electrónicos Gateway

Dispositivo	Modelo	Cantidad	Costo (\$)
Microcontrolador	Raspberry Pi 3 Mode b	1	35.00
Transceptor LoRa	RAK2245	1	120.00
Antena 915 MHz	RAK Antenna	1	27.31
UPS	DD05CVSB_5V	1	3.18
Protección batería	DD05CVSA	1	0.95
Fuente AC/DC	PSK-S10C-5-T	1	17.11
Batería	906090	1	12.00
Tarjeta PCB 60 x36 mm	ALLPCB	1	1
Costo Total			216.55

6.1.2. Componentes Mecánicos

6.1.2.1. Componentes

Nodo Sensor

En la tabla 6.3 se muestra el costo de los componentes mecánicos de 1 Nodo Sensor.

Tabla 6.3 Costo de componentes mecánicos Nodo Sensor

Dispositivo	Modelo	Cantidad	Costo (\$)
Gabinete Rittal	EB 1546.500	1	37.72
Elemento de sujecion	SZ 2584.000	1	37.63
Filtro cortaviento	SB-01	1	5.69
Costo Total			81.04

Gateway

En la tabla 6.4 se muestra el costo de los componentes mecánicos de 1 Gateway.

Tabla 6.4 Costo de componentes mecánicos Gateway

Dispositivo	Modelo	Cantidad	Costo (\$)
Gabinete Rittal	PK 9517.100	1	30.86
Elemento de sujeción	PK 9583.000	4	4.88
Costo Total			35.74

6.1.2.1. Mecanizado

Nodo Sensor

En la tabla 6.5 se muestra el costo del mecanizado de los componentes de 1 Nodo Sensor.

Tabla 6.5 Costo del mecanizado Nodo Sensor

Dispositivo	Mecanizado	Cantidad	Costo (\$)
Gabinete Rittal	Agujeros	7	3
Plancha de acero (sensor de lluvia y micrófono)	Corte	2	1.5
	Agujeros	8	1.5
	Doble	1	1.5
Tubo EMT (Microfono)	Corte	1	1.5
	Soldadura	1	4.5
Plancha de montaje	Agujeros	12	4.5
Costo Total			18

Gateway

En la tabla 6.6 se muestra el costo del mecanizado de los componentes de 1 Nodo Sensor

Tabla 6.6 Costo del mecanizado Gateway

Dispositivo	Mecanizado	Cantidad	Costo (\$)
Gabinete Rittal	Agujeros	2	1
Plancha de montaje	Agujeros	10	4
Costo Total			5

6.1.3. Caso de estudio para implementación en campo del sistema diseñado

Para esta implementación se asume que la Municipalidad de Miraflores desea monitorear la contaminación acústica en su distrito, por los alrededores del parque Kennedy debido a la gran cantidad de turistas que circulan, los cuales se llevarían una mala impresión del distrito si no se controlase bien la contaminación sonora afectando a los comercios locales.

Gateways:

Para este sistema se situarán 2 Gateway ubicados en los pisos superiores del Hotel las Américas y el edificio ABC. La altitud y ubicación de estos edificios permiten una cobertura completa a la zona de monitoreo. La señal Lora tiene un alcance de hasta al menos 3 km en zonas urbanas pero este valor se ve influenciado por la ubicación y la calidad de la antena, por seguridad se considerará un radio de solo 1.5 Km. Adicionalmente, ya que los Gateway se encuentran en zonas separadas de la ciudad, un corte eléctrico local difícilmente inhabilitará ambos Gateways a la vez. En la figura 6.1 se puede apreciar la cobertura de los Gateway, el parque Kennedy se encuentra ubicado en la intersección de estos.

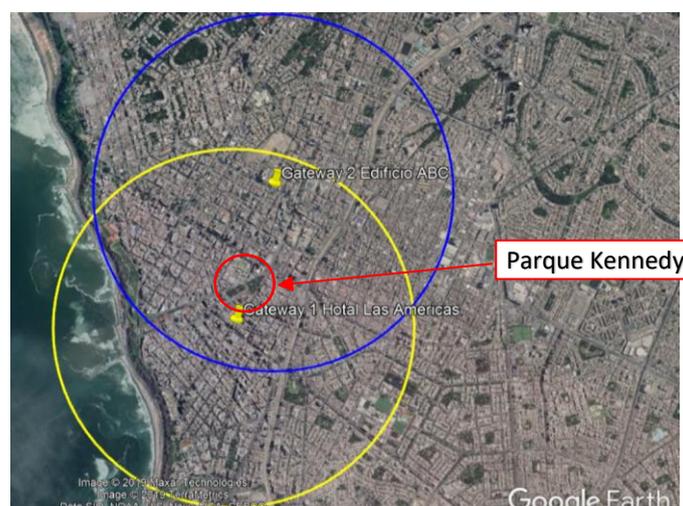


Figura 6.1 Cobertura de Gateways

Fuente: Google Earth

Nodos Sensores:

Para este sistema se situarán 18 Nodos Sensores ubicados en postes de alumbrado público en las avenidas Larco, Benavides, Shell, Diagonal, etc. Esta distribución se muestra en la figura 6.2 y le permitirá al usuario identificar los puntos críticos de contaminación.

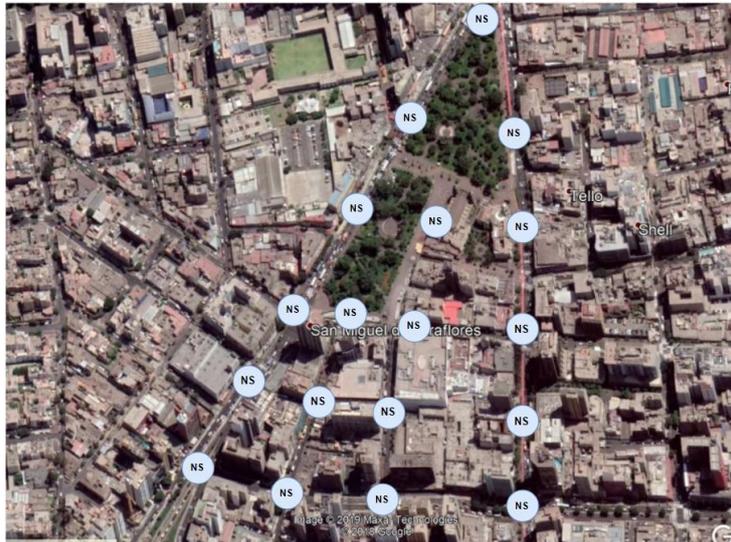


Figura 6.2 Distribución de Nodos Sensores

Fuente: Google Earth

Con la información presentada el costo de la implementación del sistema físico sería el siguiente:

$$2 \times Gateway + 18 \times NS = 2 \times (215.55 + 35.74 + 5) + 18 \times (70.57 + 81.04 + 18) = 3565.56$$

El costo total sería de \$3565.56, a esto se le agregaran 100 dólares de costos de importación de algunos componentes electrónicos y PCBs dando un costo total del **\$3665.56**.

6.2. Costo del sistema Ciber

Antes de iniciar es necesario saber que los costos de los servicios de AWS varían por región, en este caso se planea utilizar la región EE. UU Este (Ohio) por lo que los cálculos de harán basados en esas tarifas. Además, el costo será obtenido de forma anual basada en la implantación del sistema físico mencionada anteriormente y tomando de referencia los ejemplos de cálculos de precios de la página de AWS.

6.2.1 AWS IoT

Este servicio de AWS se encargará de transmitir la información enviada por los nodos sensores desde la red TTN a los correspondientes subsistemas de la Nube. Los costos de este servicio están divididos por conectividad, mensajería, registro y sombra, motor de reglas.

Conectividad

En la figura 6.3 se ve la lista de precios por millón de minutos de conexión

EE.UU. Este (Norte de Virginia)	EE.UU. Este (Ohio)	EE.UU. Oeste (Oregón)	AWS GovCloud (EE.UU.)	UE (Irlanda)
0,080 USD	0,080 USD	0,080 USD	0,100 USD	0,080 USD

Figura 6.3 Costos por conectividad AWS

Fuente: [38]

Para nuestro sistema se tendrán 18 dispositivos conectados a AWS IoT Core por 1 año por lo que siguiendo la tarifa mostrada en la figura 6.3, el costo por conectividad será el siguiente:

- Minutos de conexión = 18 conexiones * 60 minutos/hora * 24 horas/día * 365 días = 9 460 800 minutos de conexión
- Cargos totales de conectividad = 9 460 800 minutos de conexión * 0,08 USD/1 000 000 minutos de conexión = **0.76 USD**

Mensajería

En la figura 6.4 se ve la lista de precios por millón de mensajes

Volumen mensual de mensajes	EE.UU. Este (Norte de Virginia)	EE.UU. Este (Ohio)	EE.UU. Oeste (Oregón)
Hasta mil millones de mensajes	1,00 USD	1,00 USD	1,00 USD
Siguientes 4 mil millones de mensajes	0,80 USD	0,80 USD	0,80 USD
Más de 5 mil millones de mensajes	0,70 USD	0,70 USD	0,70 USD

Figura 6.4 Costos por mensajería AWS

Fuente: [38]

Este costo es bastante variable ya que la cantidad de data enviada al sistema ciber dependerá de si se dan o no condiciones mencionadas en el diagrama de flujo, para este caso se asumirá que hubo envío de data 1 vez por minuto a AWS el 60% del año. Además 1 vez al día por todo el año se envió data desde AWS hacia los nodos sensores para revisar su estado. Para el cálculo se tomará de referencia el ejemplo señalado por Amazon [38].

Costo de publicación en AWS IoT Core

- Cálculo de mensajes publicados: $18 \text{ mensaje/minuto} * 60 \text{ minutos/hora} * 24 \text{ horas/día} * 365 \text{ días} * 0.6 = 5\,676\,480 \text{ mensajes}$
- Cargos por mensajes publicados: $5\,676\,480 \text{ mensajes} * 1 \text{ USD}/1\,000\,000 \text{ mensajes} = \mathbf{5.68 \text{ USD}}$

Costo de entrega a dispositivos

- Cálculo de mensajes entregados: $18 \text{ mensaje/día} * 365 \text{ días} = 6\,570 \text{ mensajes}$
- Cargos por mensajes entregados: $6\,570 \text{ mensajes} * 1 \text{ USD}/1\,000\,000 \text{ mensajes} = \mathbf{0.00657 \text{ USD}}$

Cargos totales de mensajería

- Cargos totales de mensajería = cargos de mensajes publicados + cargos de mensajes entregados
- Cargos totales de mensajería = $5.68 \text{ USD} + 0.00657 \text{ USD} = \mathbf{5.69 \text{ USD}}$

Registro y sombra de dispositivos

En la figura 6.5 se ve la lista de precios por millón de operaciones

EE.UU. Este (Norte de Virginia)	EE.UU. Este (Ohio)	EE.UU. Oeste (Oregón)	AWS GovCloud (EE.UU.)
1,25 USD	1,25 USD	1,25 USD	1,50 USD

Figura 5.5 Costos por registro y sombra AWS

Fuente: [38]

La sombra de dispositivos se usa como medio para enviar órdenes a los nodos sensores desde el sistema ciber, en nuestro caso se utilizará únicamente para comprobar el estado de los sensores y se ha definido que esta revisión se hará 1 vez al día:

- Cargos totales de sombra de dispositivos = 18 operaciones/día * 365 días * 1,25 USD/1 000 000 operaciones = **0.0082 USD**

Motor de reglas

En la figura 6.6 se ve la lista de precios por millón de reglas disparadas / millón de acciones ejecutadas

	EE.UU. Este (Norte de Virginia)	EE.UU. Este (Ohio)	EE.UU. Oeste (Oregón)	AWS GovCloud (EE.UU.)
Reglas disparadas	0,15 USD	0,15 USD	0,15 USD	0,18 USD
Acciones ejecutadas	0,15 USD	0,15 USD	0,15 USD	0,18 USD

Figura 6.6 Costos por motor de reglas AWS

Fuente: [38]

Para este sistema, por cada mensaje enviado se dispararán 2 reglas, la primera identificará la llegada de información sobre presión sonora y ejecutará 2 acciones (envío a DynamoDB y envío a Amazon S3), la segunda identificará la llegada de información sobre fallas en los sensores y ejecutará 1 acción.

Debido a la estructura de envío de datos del sistema, no siempre se ejecutarán las acciones provenientes de las reglas porque a veces la data recibida será únicamente de presión sonora o únicamente de fallas en los sensores. Sin embargo, el cobro mínimo de AWS es de 1 acción por regla. Tomando eso en cuenta se estima que el 90% de los mensajes enviados cumplirán la primera regla, el costo de la segunda regla no cambiará debido al cobro mínimo mencionado.

- Cargos de reglas = 5 676 480 mensajes * 2 reglas/mensaje * 0,15 USD /1 000 000 de reglas disparadas = 1.70 USD
- Cargos de acciones regla 1 = 5 676 480 reglas disparadas * (2 acciones/regla * 0.9 + 1 acción/regla * 0.1) * 0,15 USD/1 000 000 de acciones ejecutadas = 1,62 USD

- Cargos de acciones regla 2 = 5 676 480 reglas disparadas * 1 acción/regla * 0,15 USD/1 000 000 de acciones ejecutadas = 0,85 USD
- Cargos totales del motor de reglas = cargos de reglas + cargos de acciones = 1,70 USD + 1.62 USD + 0.85 USD = **4.17 USD**

Finalmente, el costo anual total del servicio AWS IoT sería de **10.63 USD**

6.2.2 DynamoDB

Este servicio de AWS tiene una capa gratuita de por vida que Brinda hasta 25GB de almacenamiento y 200 millones de solicitudes (lectura o escritura) al mes, de superarse este monto el costo del servicio seria el que se muestra en la figura 6.7.

Unidades de solicitud de escritura	1,25 USD por millón de unidades de solicitud de escritura
Unidades de solicitud de lectura	0,25 USD por millón de unidades de solicitud de lectura

- Los primeros 25 GB almacenados cada mes son gratis
- Los siguientes, 0,25 USD por GB al mes

Figura 6.7 Costos de DynamoDB
Fuente: [30]

El sistema implementado realizará una solicitud de escritura por cada dato de nivel de presión sonora que obtenga de los sensores, además la función lambda realizará una solicitud de lectura cada vez que un usuario lo solicite para obtener la media de las medidas de cada punto en los últimos 10 min y una lectura por escritura para la central.

En este cálculo de debe considerar que una unidad de escritura ocupa hasta 1KB, una unidad de lectura hasta 8KB, el peso estimado de cada data recibida es de 1KB y que el 90% de la data son presiones sonoras.

- Escritura = 5 676 480 mensajes * 0.9 * 1 KB/mensaje * 1 escritura/KB = 5 108 832 solicitudes de escrituras anuales.
- Almacenamiento = 5 676 480 mensajes * 0.9 * 1 KB/mensaje * 1 GB/1000000 KB = 5.12 GB anuales.

Para la lectura el costo es bastante variable ya que dependerá del uso del App por parte de los ciudadanos y la interface web por parte la municipalidad. Considerando que Miraflores tiene una población de aproximadamente 100 mil habitantes y el uso de este tipo de apps no es muy frecuente, se estima una cantidad diaria de 5 000 usos. Por otro

lado, se considerará una central de monitoreo que recolectará la data de todos los puntos en todo momento.

- Lectura por App = 5 000 usos/día * 365 días * 10 mediciones/uso * 1 KB/medición * 1 lectura/8 KB = 2 281 250 solicitudes de lectura anuales.
- Lectura de central = 5 676 480 mensajes * 1 lectura/mensaje = 5 676 480 solicitudes de lectura anuales.

Con los cálculos anuales obtenidos se aprecia claramente que la capa gratuita mensual no es superada por lo que el costo de este servicio será de **0 USD**.

6.2.3 Amazon S3

En la figura 6.8 se ve la lista de precios de almacenamiento por GB de Amazon S3

Almacenamiento estándar en S3	
Primeros 50 TB/mes	0,023 USD por GB
Siguientes 450 TB/mes	0,022 USD por GB
Más de 500 TB/mes	0,021 USD por GB

Figura 6.8 Costos de Amazon S3

Fuente: [31]

Ya que la data almacenada en esta base de datos será acumulativa y no se borrará, se asumirá un aumento lineal de la data hasta llegar a los 5.12 GB almacenados al final del primer año dando como resultado la siguiente fórmula:

- Consumo mensual = 5.12 GB/12 meses * número de meses de almacenamiento
- Costo mensual = consumo mensual * 0.023

Para tener el costo del primer año bastará con calcular el área bajo la curva de la función.

- Costo del primer año de almacenamiento = $(12 * 5.12 * 0.023)/2 = \mathbf{0.71 USD}$

6.2.4 AWS Lambda

Amazon lambda posee una capa gratuita de 1 millón de solicitudes y 400 000 GB de tiempo de cómputo al mes [32], para el sistema planteado el consumo sería el siguiente:

- Solicitudes de App al mes = 1 solicitud/uso * 5000 usos/día * 30 días = 150 000 solicitudes
- Solicitudes web al mes = 18 solicitudes/minuto * 1440 minutos/día * 30 días = 777 600 solicitudes

A estas solicitudes se les de agregar las causadas por eventos de mal funcionamiento de sensores, pérdida de energía y la solicitud en caso de recopilación de toda la data. Sin embargo, estas solicitudes son en mucha menor cantidad y no se superará el límite gratuito.

Ya que la función lambda más compleja solo lee los últimos valores y los promedia para mostrar al usuario, el tamaño y tiempo de cómputo es mínimo y no se superará la capa gratuita. Esta información más la obtenida de solicitudes por mes nos da un costo de **0 USD**

6.2.5 AWS CloudFront

El costo de este servicio funciona de manera similar a Amazon S3 sin embargo el espacio utilizado es de transferencia y no de almacenamiento de datos. Al trabajar ambos servicios con la misma data, el cálculo del costo será el mismo que Amazon S3 y solo habrá que cambiar la tarifa siguiendo la mostrada en la figura 6.9.

al mes	Estados Unidos y Canadá	Europa	Sudáfrica y Oriente Medio
Primeros 10 TB	0,085 USD	0,085 USD	0,110 USD
Siguientes 40 TB	0,080 USD	0,080 USD	0,105 USD
Siguientes 100 TB	0,060 USD	0,060 USD	0,090 USD

Figura 6.9 Costos de Amazon CloudFront

Fuente: [34]

- Costo del primer año de uso = $(12 * 5.12 * 0.085) / 2 = \mathbf{2.61 USD}$

6.2.6 API Gateway

El costo de este servicio es por millón de solicitudes como se muestra en la figura 6.10

Número de solicitudes (por mes)	Precio (por millón)
Primeros 333 millones	3,50 USD
Próximos 667 millones	2,80 USD
Próximos 19 mil millones	2,38 USD
Más de 20 mil millones	1,51 USD

Figura 6.10 Costos de API Gateway

Fuente: [33]

En el sistema analizado, Api Gateway atenderá las solicitudes de los ciudadanos y la municipalidad para obtener el nivel de presión sonora en tiempo real por lo que el número de solicitudes será el mismo que el calculado para el servicio lambda:

- Costo anual = 927 600 solicitudes/mes * 12 meses * 3.5 USD/1 000 000 solicitudes = **38.96 USD**

6.2.7 Amazon Cognito

Este servicio tiene una capa gratuita de 50 000 usuarios [35] y ya que el control de acceso solo será aplicado en usuarios administradores, los cuales no superaran en número a la capa gratuita, el costo será de **0 USD**.

6.2.8 Amazon CloudWatch

Ya que el sistema a implementar de tamaño reducido y estructura poco compleja, se utilizará únicamente la capa gratuita que ofrece este servicio la cual cubre las características mostradas en la figura 6.11.

Métricas	Métricas de monitoreo básico (frecuencia de 5 minutos)
	10 métricas de monitoreo detallado (frecuencia de 1 minuto)
	1 millón de solicitudes API (no aplicable a GetMetricData ni GetMetricWidgetImage)
Panel	3 paneles para hasta 50 métricas al mes
Alarmas	10 métricas de alarma (no se aplica a alarmas de alta resolución)
Logs	5 GB de datos (incorporación, almacenamiento y archivo, y datos escaneados por las consultas de Logs Insights)
Eventos	Se incluyen todos los eventos excepto los personalizados

Figura 6.11 Capa gratuita Amazon CloudWatch

Fuente: [36]

Con toda la información recolectada se obtiene que el costo de 1 año de implementación de la arquitectura Cloud del sistema ciber costaría aproximadamente **\$52.91**. En la tabla 6.7 se muestra un resumen de los costos calculados.

Tabla 6.7 Costo anual del sistema Ciber

Servicio	Costo anual (\$)
AWS IoT	10.63
DyanmoDB	0
Amazon S3	0.71
AWS Lambda	0
AWS CloudFront	2.61
API Gateway	38.96
Amazon Cognito	0
Amazon CloudWatch	0
Costo Total	52.91

6.3. Costo total del sistema

El resumen de los costos del sistema ciberfísico en la implementación de un sistema de monitoreo de contaminación acústica para los alrededores del parque Kennedy es el siguiente:

- Costo del sistema físico: 3665.56 USD
- Costo del sistema ciber: 52.91 USD
- Costo del sistema ciberfísico: $3665.56 + 52.91 = 3718.47$ USD

A este resultado se le adicionará el costo de diseño, prototipo y pruebas el cual resulta en 3200 USD ya que el tiempo de diseño de este sistema fue de aproximadamente 4 meses y se estima un costo mensual de trabajo de 800 USD.

- Costo total del sistema: $3718.47 + 3200 = \mathbf{6919}$ USD

CONCLUSIONES Y RECOMENDACIONES

- En la presente tesis se logró diseñar un sistema de monitoreo de contaminación acústica urbana que funciona con radiofrecuencia y bajo una plataforma IoT; este dispositivo utiliza la tecnología de radiofrecuencia LoRa la cual sustituye la comunicación directa a internet comúnmente utilizada en este tipo de aplicaciones. Gracias a esto, se presenta una solución novedosa de bajo consumo energético y computacional capaz de cubrir grandes zonas de la ciudad y enviar la data a una plataforma Cloud sin necesidad de depender de redes WIFI, la cual podrá ser implementada con el objetivo de monitorear y controlar este tipo de contaminación de forma más eficaz.
- Según los cálculos realizados y basándonos en los requerimientos y especificaciones técnicas planteadas, los dispositivos de monitoreo son capaces de funcionar de forma continua a pesar de cortes temporales de energía o condiciones climáticas desfavorables gracias a la batería de respaldo y los sensores de lluvia y corriente instalados. Además, la instalación y mantenimiento de los mismos es bastante sencilla lo cual permite una rápida escalabilidad.
- Se logró implementar un prototipo del sistema con el que se pudo probar la conectividad entre dispositivos, la capacidad del nodo sensor para capturar y procesar datos de contaminación sonora y la transmisión de los mismos hacia un servidor Cloud. Además, se realizó una comparación de las mediciones con un sonómetro profesional y el prototipo donde tras una calibración dio valores que, si bien no logran la precisión del primero, cumplen con los requerimientos del proyecto.
- El costo total del sistema para el ejemplo de implementación es de 6919 dólares. Sin embargo, este costo-beneficio mejora mientras mayor sea el escalamiento y tiempo de operación ya que los gastos energéticos y de sostenimiento web son muy reducidos.

- Como recomendación para mejorar el trabajo realizado quedaría pendiente la implementación de la arquitectura de software y el diseño una interfaz de usuario que muestre correctamente la información recolectada y procesada por el sistema. Además, se podría implementar un prototipo más completo de la parte física con el cual se puedan hacer pruebas de campo para verificar la existencia interferencias con la captura de datos que no hayan sido consideradas durante el desarrollo del sistema.
- El trabajo realizado se centra en el monitoreo de la contaminación acústica urbana; sin embargo, manteniendo como base la arquitectura IoT y la comunicación mediante red LPWAN, se podría ampliar el alcance a un diseño más complejo que cubra además otros problemas de monitoreo urbano como la contaminación del aire o la temperatura del ambiente.

BIBLIOGRAFÍA

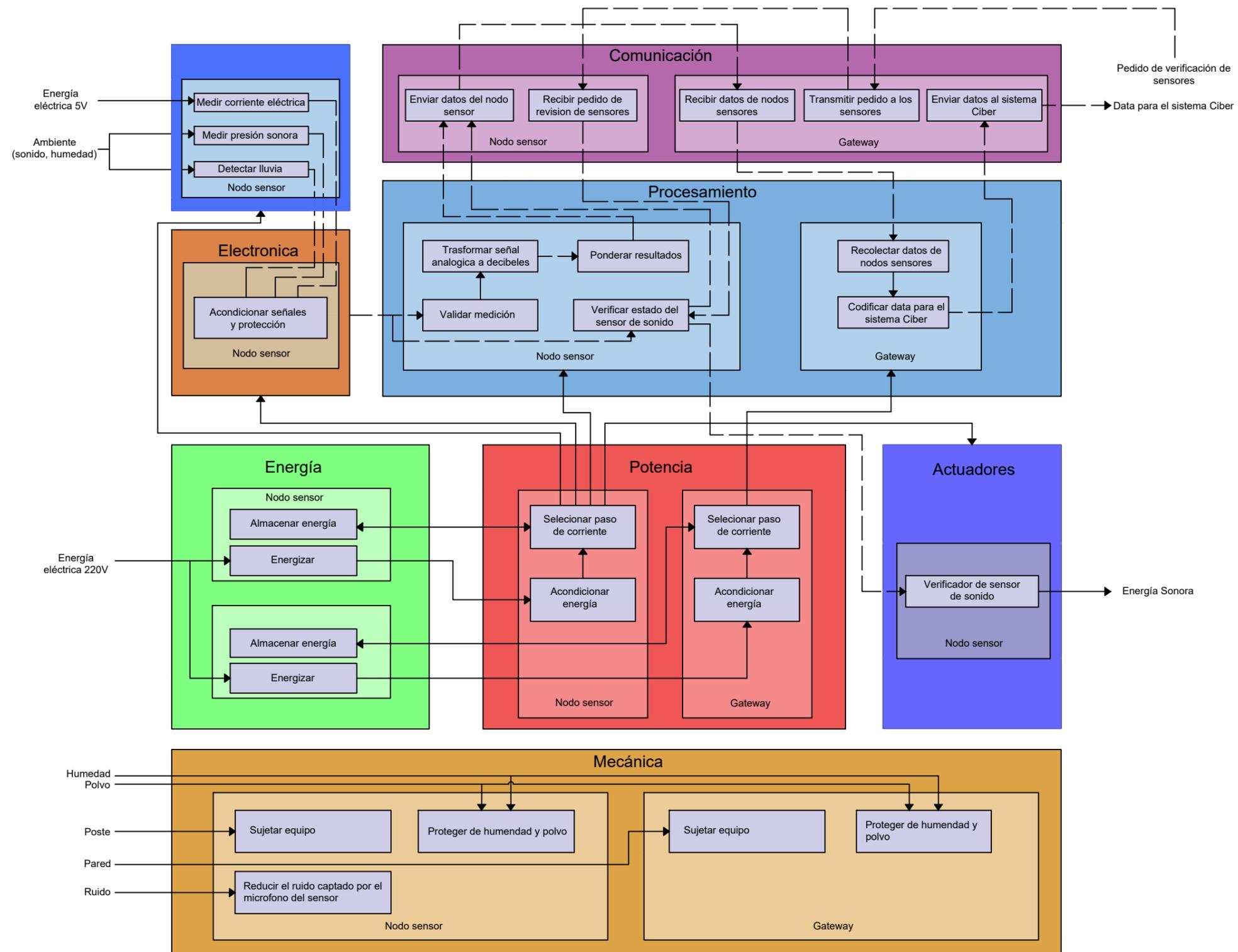
- [1] Organismo de Evaluación y Fiscalización Ambiental, «La contaminación sonora en Lima y Callao,» Oficina de Comunicaciones y Atención al Ciudadano, Lima, 2016.
- [2] D. G. ALBERT y S. N. DECATO, «Acoustic and seismic ambient noise measurements in urban and rural areas,» *Applied Acoustics - Elsevier*, vol. 119, pp. 135-143, 2017.
- [3] Diario Correo, «diariocorreo.pe,» 19 Marzo 2018. [En línea]. Available: <https://diariocorreo.pe/edicion/lima/la-contaminacion-sonora-y-los-serios-danos-que-provoca-en-tu-salud-808421/>. [Último acceso: 20 Marzo 2019].
- [4] P. E. K. FIEDLER y P. H. T. ZANNIN, «Evaluation of noise pollution in urban traffic hubs—Noise maps and measurements,» *Environmental Impact Assessment Review - Elsevier*, vol. 51, pp. 1-9, 2015.
- [5] S. Valera, «Elementos básicos de psicología ambiental,» [En línea]. Available: http://www.ub.edu/psicologia_ambiental/unidad-4-tema-9-5-3/. [Último acceso: 22 Marzo 2019].
- [6] E. Kanjo, *NoiseSPY: A Real-Time Mobile Phone Platform for Urban*, Cambridge: Springer Science, 2009.
- [7] MINAM, *Protocolo Nacional de Monitoreo Ambiental*, Lima, 2013.
- [8] A. d. E. Alonso, *Contaminación acústica y salud*, Madrid: Observatorio medioambiental, 2003.
- [9] Á. F. R. Ridao, *MEDIDAS DE RUIDO*, Granada, 2009.
- [10] A. Hidalgo, «¿Qué es el decibelio acústico?,» [En línea]. Available: <http://www.cecorsl.com/2012/11/16/que-es-el-decibelio-acustico/>. [Último acceso: 4 Abril 2019].
- [11] Cisco Website, «Cisco internet of things,» [En línea]. Available: <https://www.cisco.com/c/en/us/solutions/internet-of-things/overview.html>. [Último acceso: 23 Marzo 2019].
- [12] IoT Factory, «Overview of IOT Networks,» [En línea]. Available: <http://iotfactory.eu/iot-knowledge-center/overview-of-iot-networks/>. [Último acceso: 14 Abril 2019].
- [13] B. Cendón, «Las Redes Más Usadas En El IoT,» 30 Enero 2017. [En línea]. Available: <http://www.bcendon.com/las-redes-mas-usadas-en-el-iot/>. [Último acceso: 14 Abril 2019].
- [14] N. Aakvaag y J.-E. Frey, «Redes de sensores inalámbricos,» *Revista ABB 2*, pp. 39-42, 2006.

- [15] J. Fuller, «TechBeacon,» 26 Mayo 2016. [En línea]. Available: <https://techbeacon.com/enterprise-it/4-stages-iot-architecture>. [Último acceso: 5 Abril 2019].
- [16] R. Wojcik, F. Bachmann, L. Bass, P. Clements, P. Merson, R. Nord y B. Wood, «Attribute-driven design (ADD), version 2.0,» *CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST*, 2006.
- [17] R. Kazman y H. Cervantes, *ADD 3.0: Rethinking Drivers and Decisions in the Design Process*, SATURN, 2015.
- [18] Intelkia, «IN_NOISE – Solución IoT para la monitorización del ruido,» [En línea]. Available: <https://www.intelkia.com/productos/in-noise-solucion-iot-monitorizacion-ruido/>. [Último acceso: 5 Abril 2019].
- [19] Kunak, «KUNAK NOISE N10 Sistema de monitorización de ruido,» [En línea]. Available: <https://www.kunak.es/productos/monitorizacion-ambiental/monitor-de-ruido/>. [Último acceso: 05 Abril 2019].
- [20] Cesva, «TA120 Sensor de monitorización de ruido,» [En línea]. [Último acceso: 05 Abril 2019].
- [21] C. Mydlarz, J. Salamon y J. P. Bello, *The Implementation of Low-cost Urban Acoustic Monitoring Devices*, New York: New York University, 2016.
- [22] P. G. Sánchez, *Desarrollo de un sistema de medición de ruido acústico*, Granada: Universidad de Granada, 2016.
- [23] H. R. Altamirano Benalcázar y A. M. Puente Moreno, *Diseño e implementación de una red inalámbrica de sensores inteligentes para el registro de la contaminación acústica en la epoch basado en tarjetas de desarrollo*, Chimborazo: Escuela Superior Tecnica de Chimborazo, 2016.
- [24] A. J. Wixted, P. Kinnaird, H. T. A. Larijani, A. Ahmadinia y N. Strachan, «Evaluation of LoRa and LoRaWAN for wireless sensor networks,» *2016 IEEE SENSORS*, pp. 1-13, 2016.
- [25] E. Perez, «Medium,» 27 Agosto 2018. [En línea]. Available: <https://medium.com/pruebas-de-laboratorio-de-la-modulaci%C3%B3n-lora/lorawan-d00f48384160>. [Último acceso: 03 Mayo 2019].
- [26] Ministerio del Ambiente, *Protocolo Nacional de Monitoreo de Ruido Ambiental*, Lima, 2013.
- [27] Microsoft, *Microsoft Azure IoT Reference Architecture Version 2.1*, 2018.

- [28] The Things Network, «The Things Network.» [En línea]. Available: <https://www.thethingsnetwork.org/docs/network/architecture.html>. [Último acceso: 15 Junio 2019].
- [29] Amazon, «AWS IoT Guia de desarrollador.» [En línea]. Available: https://docs.aws.amazon.com/es_es/iot/latest/developerguide/what-is-aws-iot.html. [Último acceso: 15 Junio 2019].
- [30] Amazon, «Amazon DynamoDB.» [En línea]. Available: <https://aws.amazon.com/es/dynamodb/>. [Último acceso: 13 junio 2019].
- [31] Amazon, «Amazon S3.» [En línea]. Available: <https://aws.amazon.com/es/s3/>. [Último acceso: 14 Junio 2019].
- [32] Amazon, «AWS Lambda.» [En línea]. Available: <https://aws.amazon.com/es/lambda/>. [Último acceso: 14 Junio 2019].
- [33] Amazon, «Amazon API Gateway.» [En línea]. Available: <https://aws.amazon.com/es/api-gateway/>. [Último acceso: 14 Junio 2019].
- [34] Amazon, «Amazon CloudFront.» [En línea]. Available: <https://aws.amazon.com/es/cloudfront/>. [Último acceso: 15 Junio 2019].
- [35] Amazon, «Amazon Cognito.» [En línea]. Available: <https://aws.amazon.com/es/cognito/>. [Último acceso: 15 Junio 2019].
- [36] Amazon, «Amazon CloudWatch.» [En línea]. Available: <https://aws.amazon.com/es/cloudwatch/>. [Último acceso: 15 Junio 2019].
- [37] C. Samuelson, «Hackster.io,» 14 Enero 2017. [En línea]. Available: <https://www.hackster.io/ChrisSamuelson/lora-raspberry-pi-single-channel-gateway-cheap-d57d36>. [Último acceso: 9 Setiembre 2019].
- [38] Amazon, «Precios de AWS IoT Core.» [En línea]. Available: <https://aws.amazon.com/es/iot-core/pricing/>. [Último acceso: 16 Junio 2019].
- [39] G. B. S. A. A. S. Tatiana Aguilar, «Monitoreo y Videovigilancia basado en IoT en tiempo,» *RISTI*, n° E23, p. 288–301, 2019.

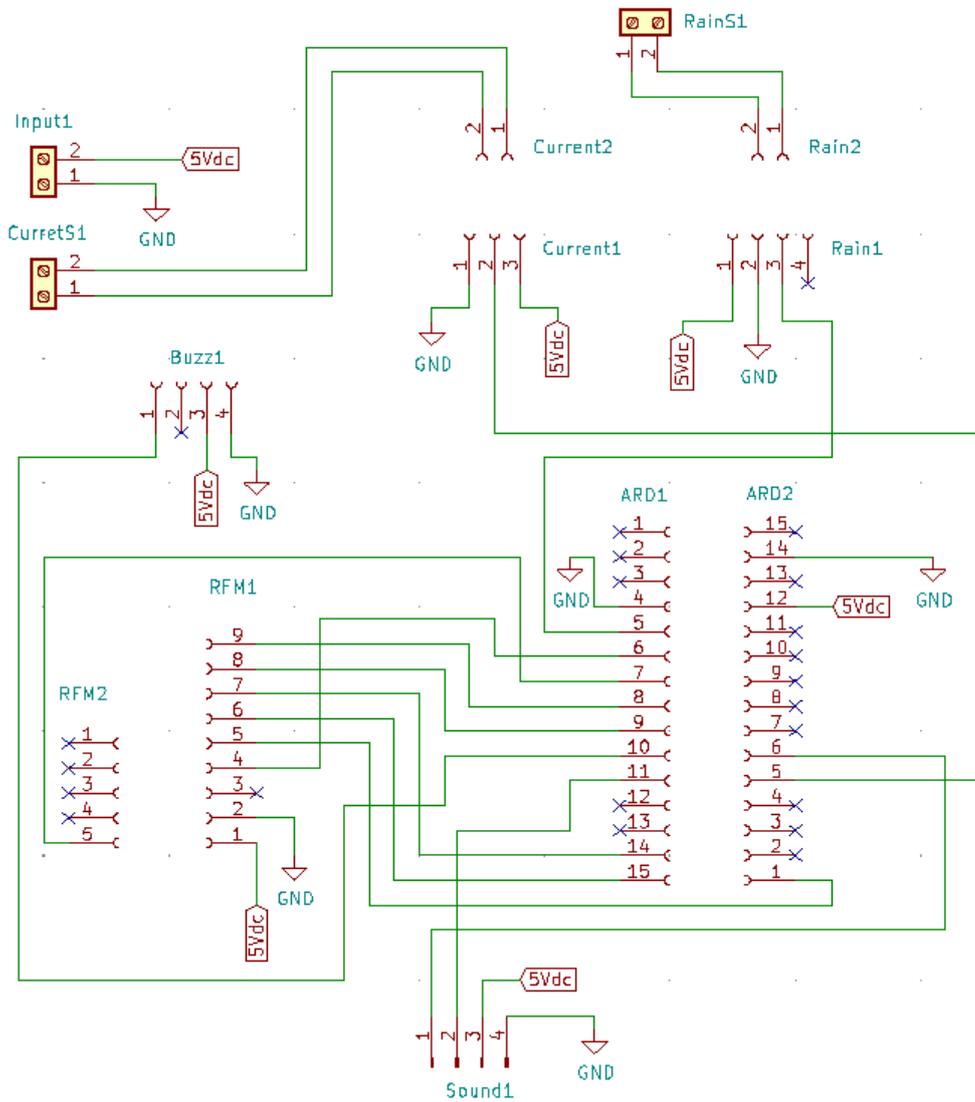
ANEXOS Y PLANOS

ANEXO 1: Estructura de funciones

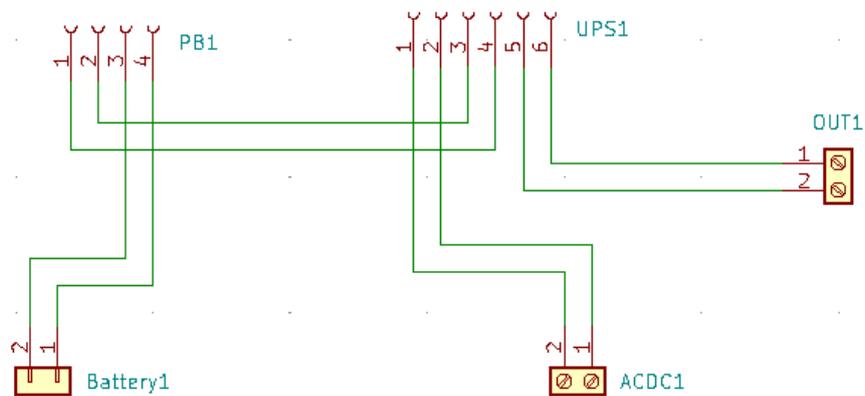


Anexo 2

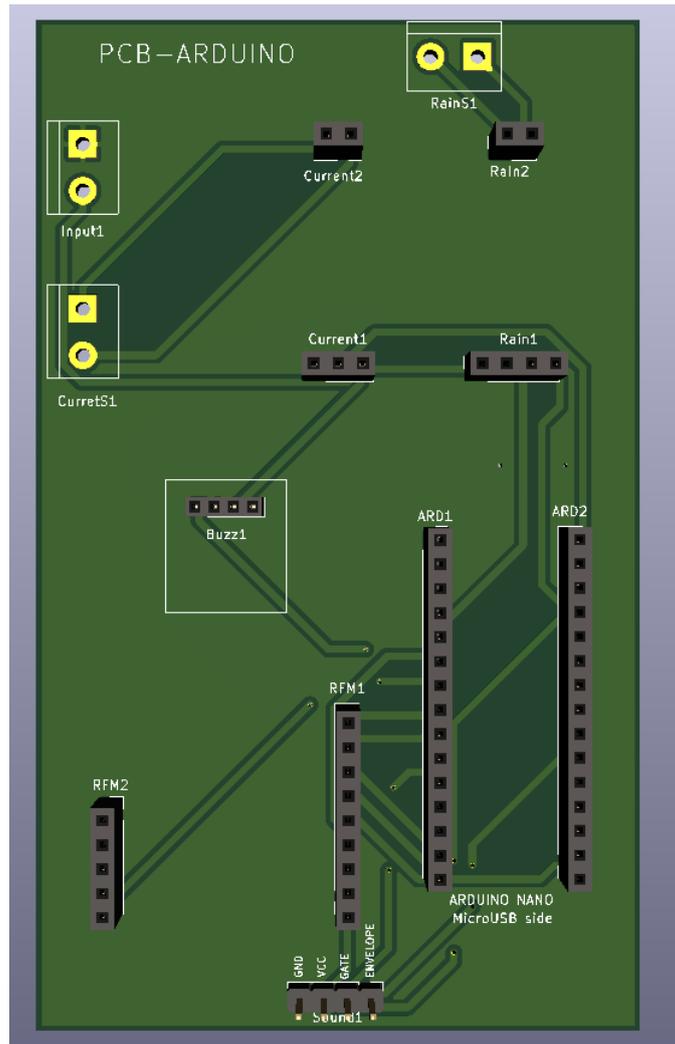
Esquemático PCB-ARDUINO



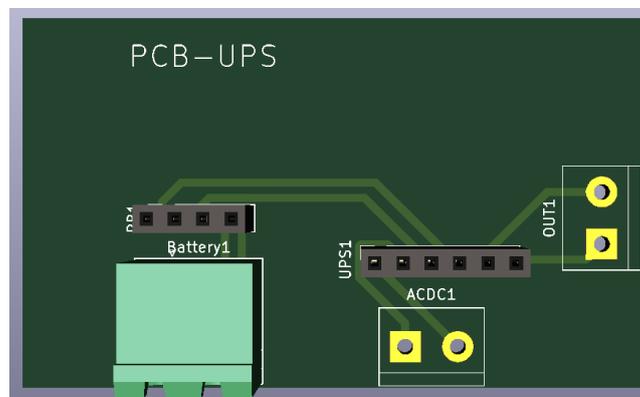
Esquemático PCB-UPS



Modelo 3D PCB-Arduino

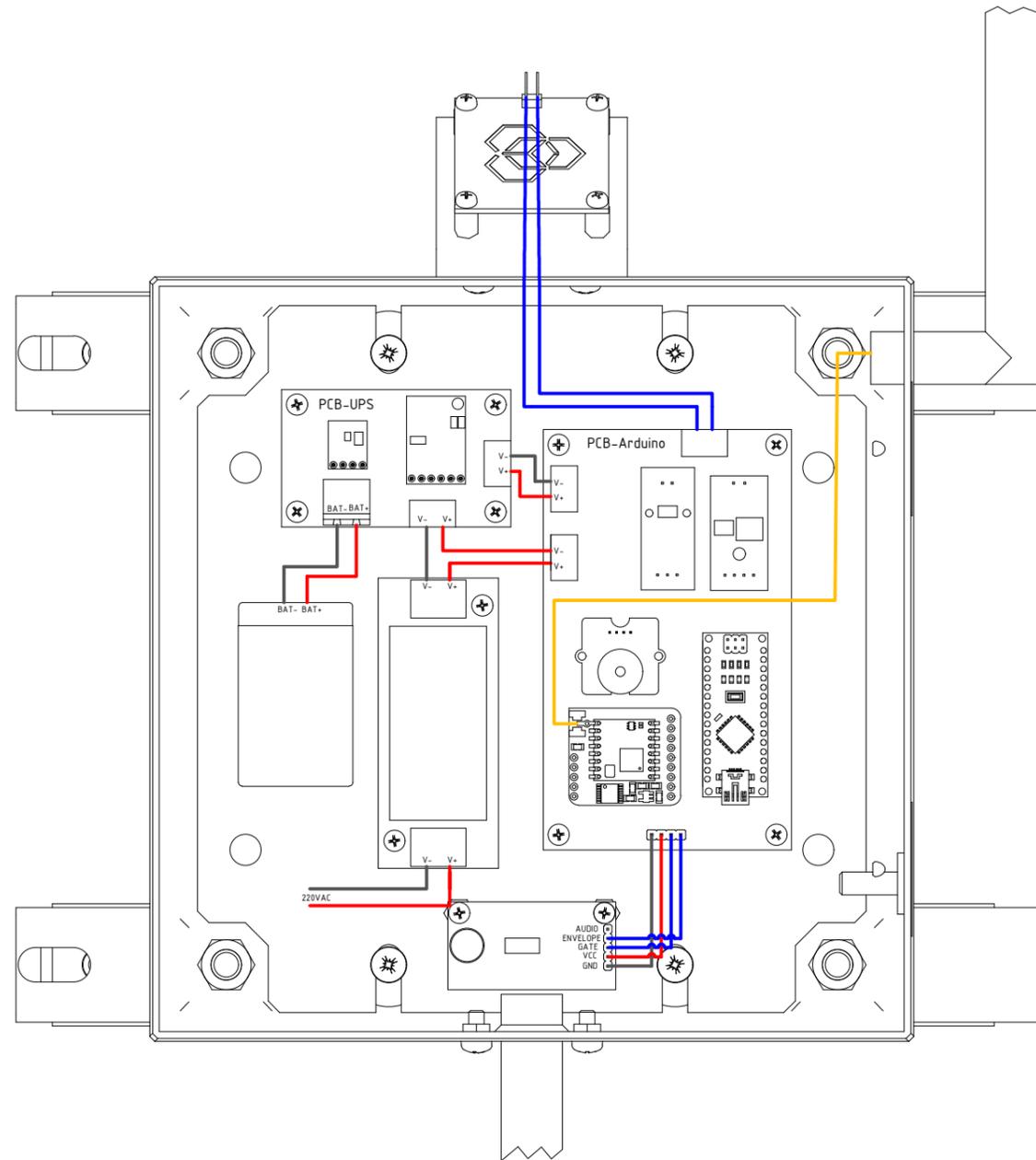


Modelo 3D PCB-UPS

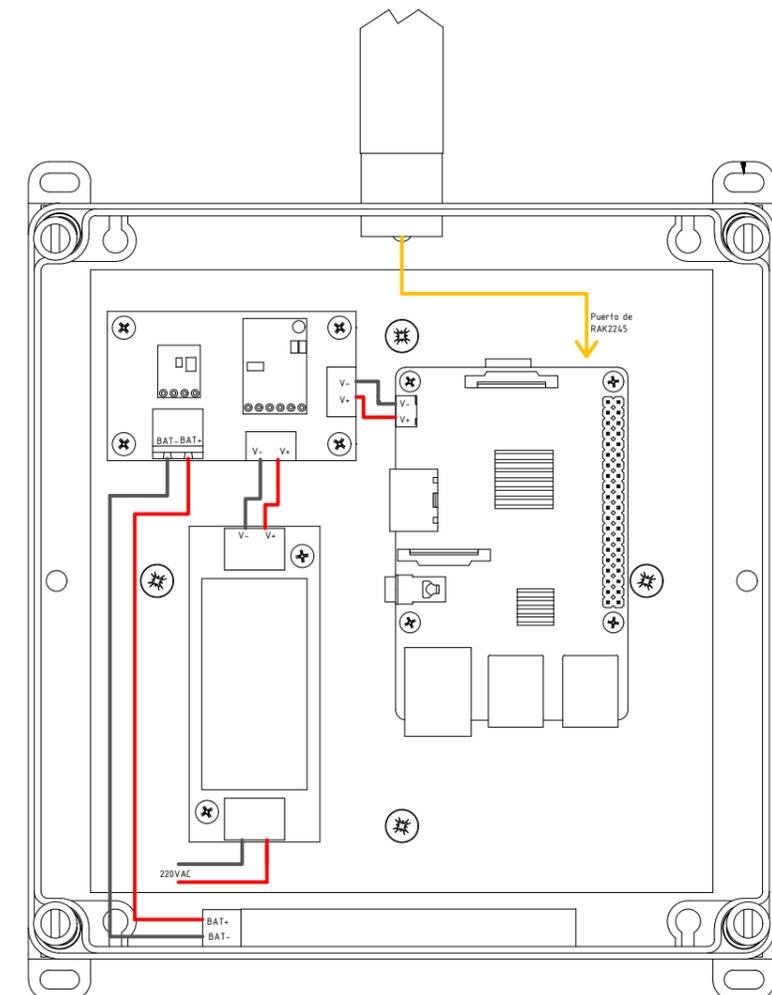


Anexo 3: Conexiones

Conexiones Nodo Sensor

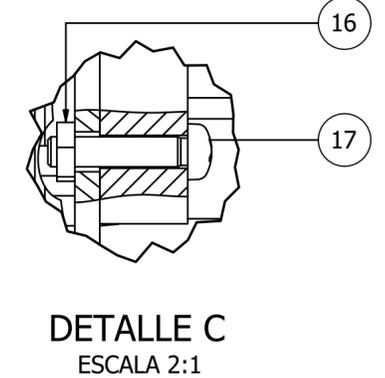
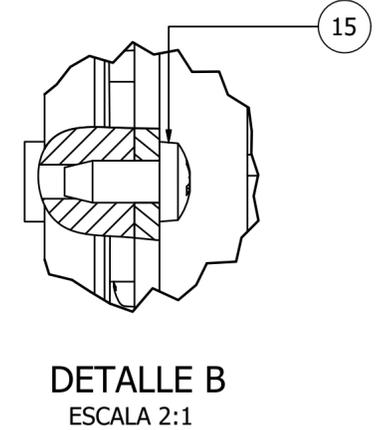
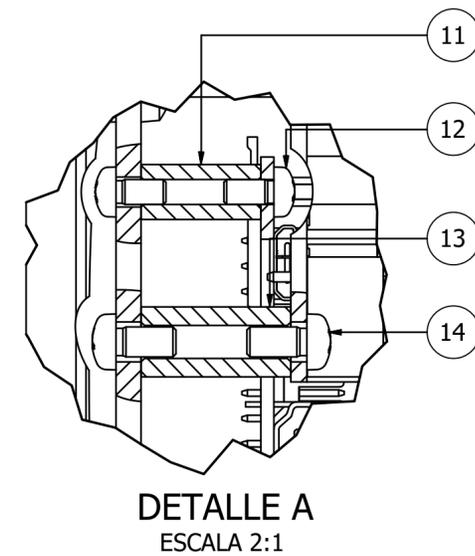
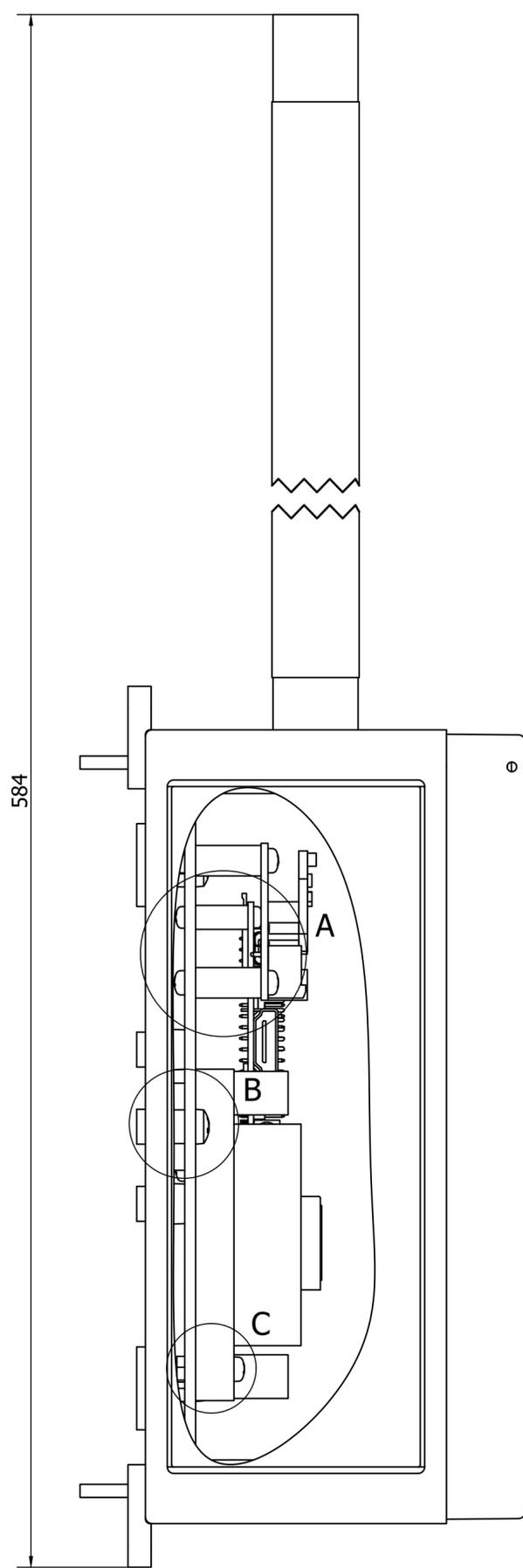
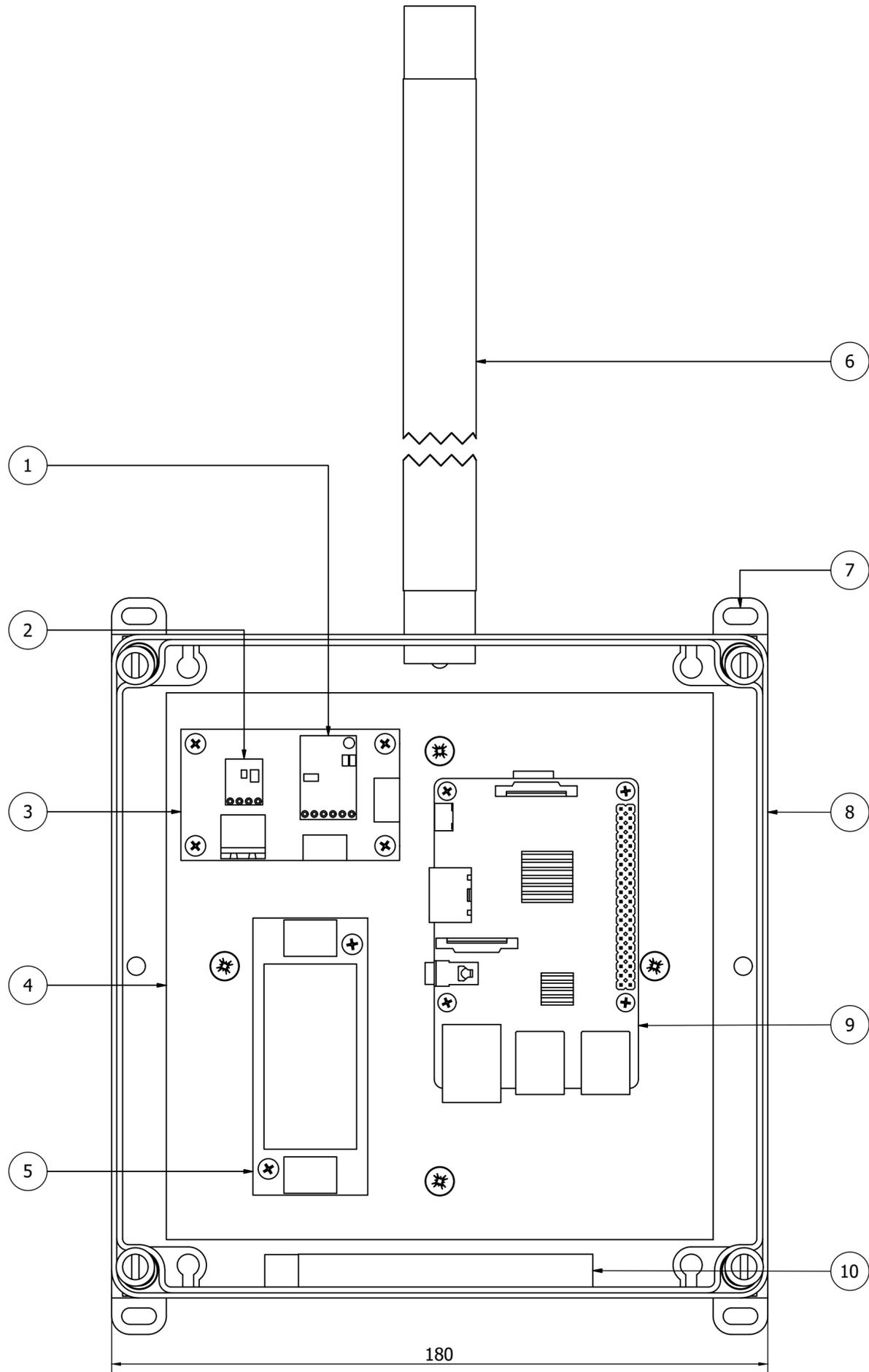


Conexiones Gateway



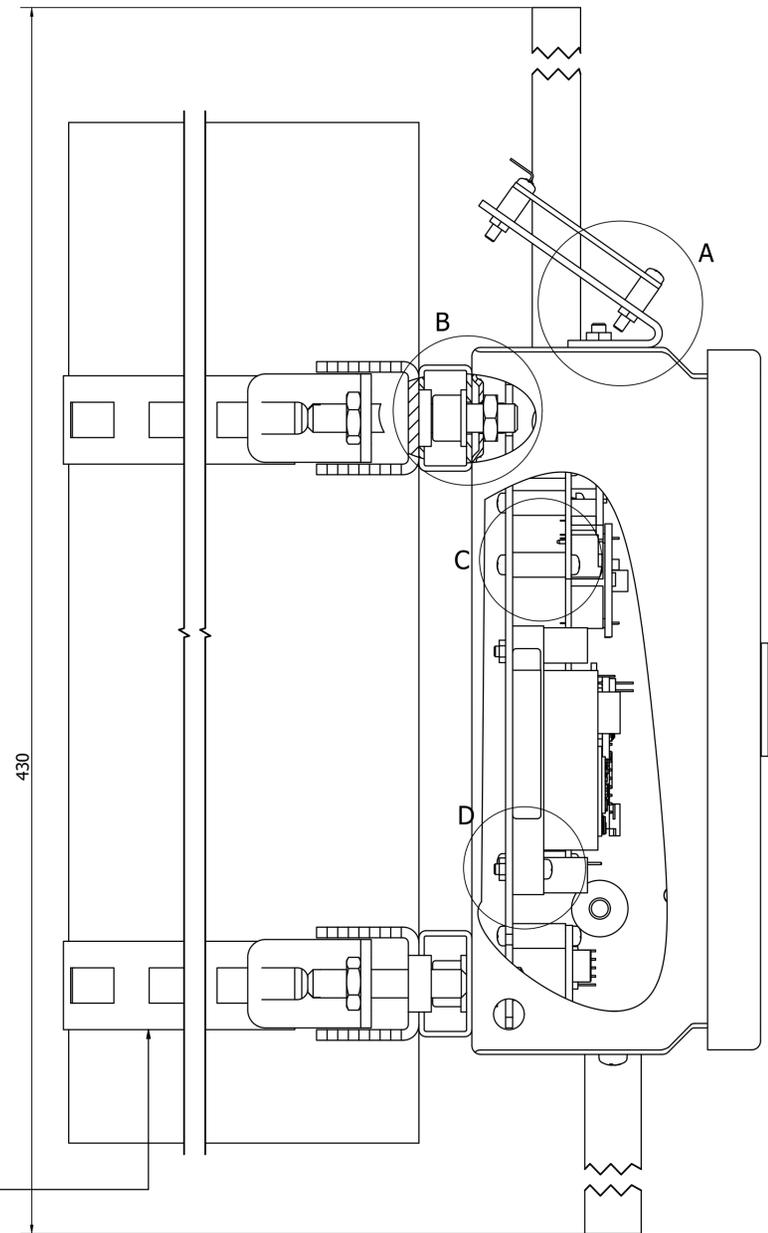
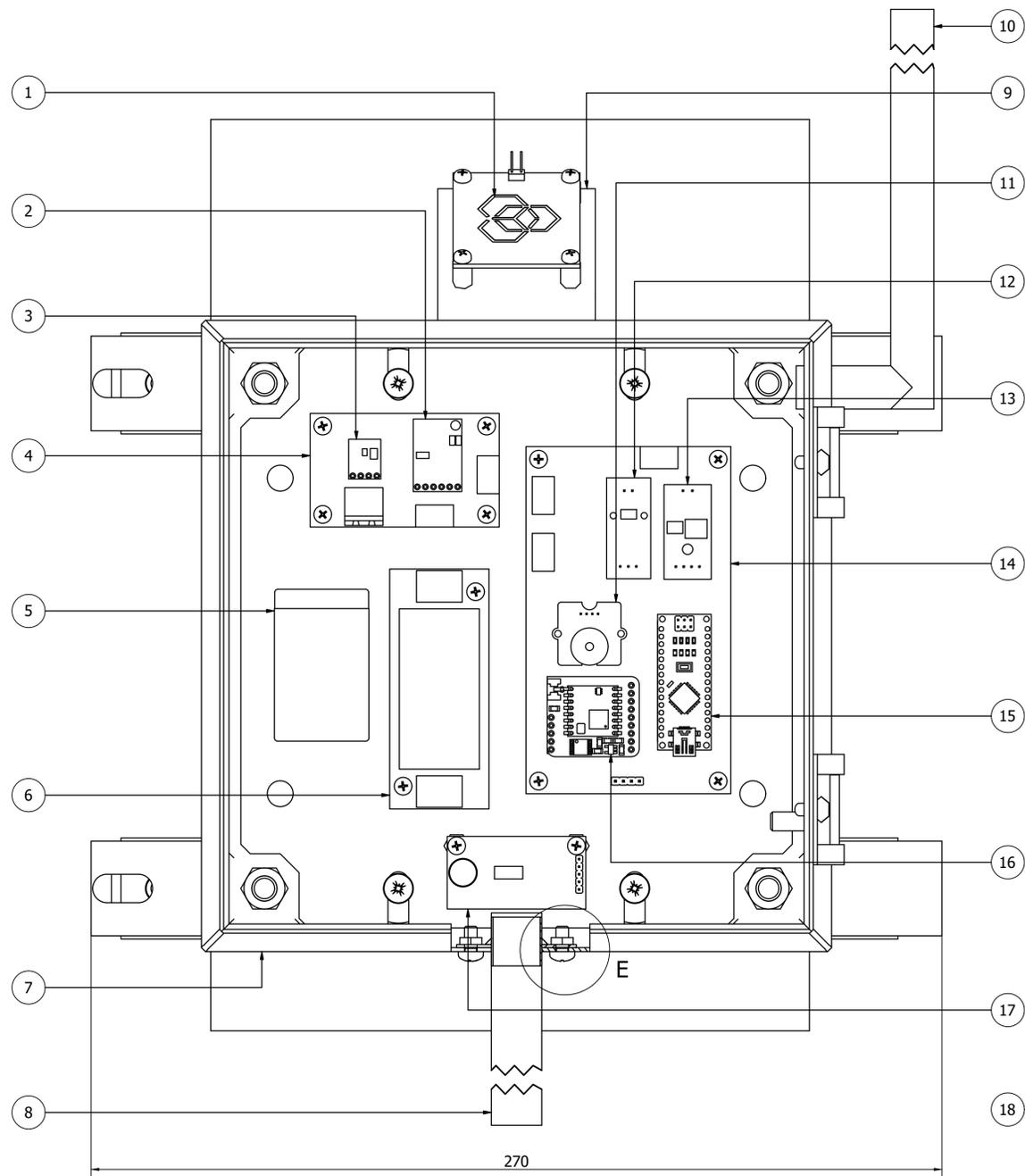
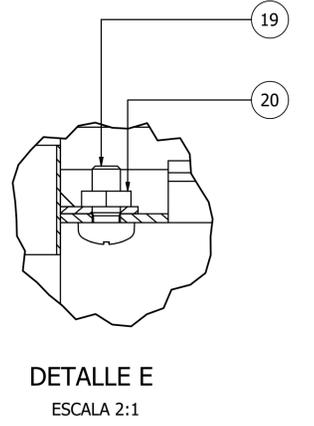
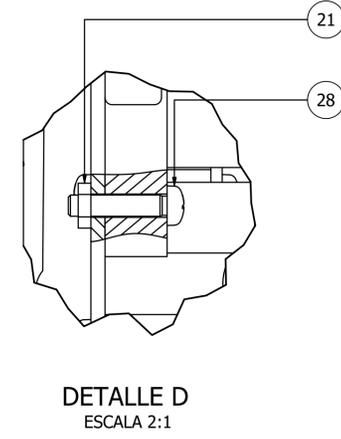
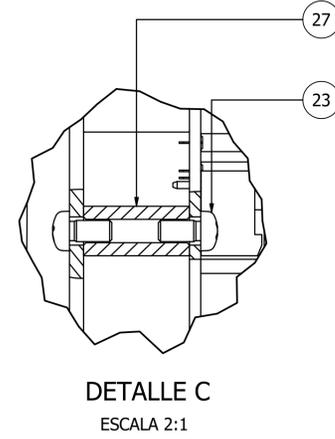
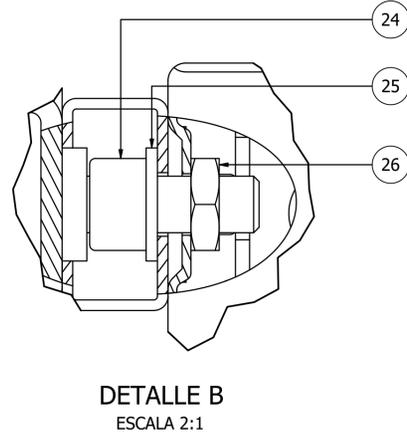
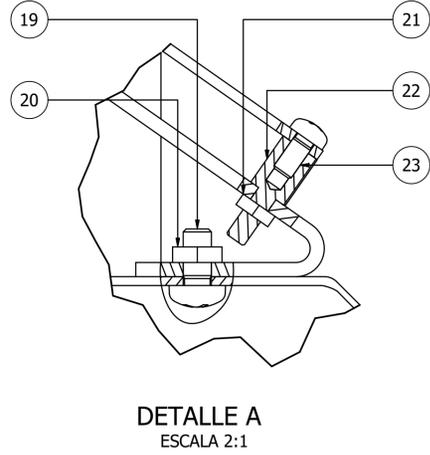
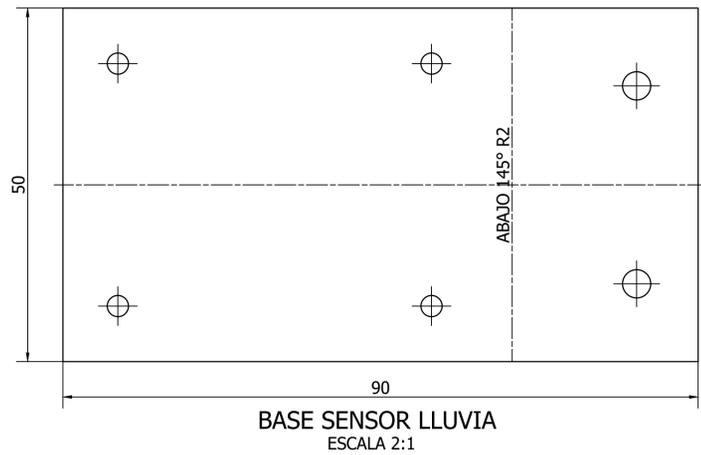
Leyenda

- Linea positiva 
- Linea negativa 
- Conexion antena 
- Otras conexiones 



17	2	TORNILLO CABEZA REDONDA	ISO7045	AISI304	M3X14-4.8-H
16	2	TUERCA HEXAGONAL	ISO4036	AISI304	M3
15	4	TORNILLO AUTORROSCANTE	ISO7049	AISI1010	ST4.2X9.5-F-Z
14	8	TORNILLO CABEZA REDONDA	ISO7045	AISI304	M3X6-4.8-H
13	4	TORNILLO ESPACIADOR		LATON CuZn38Pb2	HTSB-M3-15-5-1
12	8	TORNILLO CABEZA REDONDA	ISO7045	AISI304	M2.5X5-4.8-H
11	4	TORNILLO ESPACIADOR		LATON CuZn38Pb2	HTSB-M2.5-12-5-1
10	1	BATERIA DE LITIO 3.7V 6000mAh		VARIOS	906090
9	1	RASPBERRI PI 3 + RAK2245		VARIOS	
8	1	CAJA RITTAL		POLICARBONATO	PK 9517.100
7	4	SOPORTES DE MONTAJE EN PARED		POLIAMIDA	PK 9583.000
6	1	ANTENA DE 915MHz		VARIOS	RAK ANTENNA
5	1	FUENTE DE 5V-2A		VARIOS	PSK-S10C-5-T
4	1	PLACA DE MONTAJE		AISI1010	PK 9548.000
3	1	PCB POTENCIA		VARIOS	PCB-UPS
2	1	PROTECTOR DE BATERIA DE LITIO		VARIOS	DD05CVSA
1	1	UPS DE 5V		VARIOS	DD05CVSB
POS. CANT.		DESCRIPCION	NORMA	MATERIAL	OBSERVACIONES

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ FACULTAD DE CIENCIAS E INGENIERÍA – ESPECIALIDAD: ING. MECATRÓNICA		
METODO DE PROYECCIÓN	TRABAJO FIN DE CARRERA 2 - 10M2	ESCALA
		1:1
20142577	MURATA ARCA, RENZO MITSUO	FECHA: 2019.06.29
		LAMINA: A2



28	2	TORNILLO CABEZA REDONDA	ISO 7045	AISI304	M3X14-4.8-H
27	10	TORNILLO ESPACIADOR		LATON CuZn38Pb2	HTSB-M3-15-5-1
26	4	TUERCA HEXAGONAL	ISO 4035	AISI304	M8
25	4	ARANDELAS PLANAS	ISO 7091	AISI1010	M8
24	4	TORNILLOS HEXAGONALES DE CABEZA HUECA	ISO 4762	AISI304	M8X16
23	24	TORNILLO CABEZA REDONDA	ISO 7045	AISI304	M3X6-4.8-H
22	4	TORNILLO ESPACIADOR		LATON CuZn38Pb2	HTSB-M3-5-5-2
21	6	TUERCA HEXAGONAL	ISO 4036	AISI304	M3
20	4	TUERCA HEXAGONAL	ISO 4036	AISI304	M4
19	4	TORNILLO CABEZA REDONDA	ISO 7045	AISI304	M4X8-4.8-H
18	1	ABRAZADERA CIRCULAR		AISI1010	SZ 2584.000
17	1	SENSOR DE SONIDO	VARIOS		SEN-12642
16	1	TRANCEPTOR LORA	VARIOS		RFMS5W
15	1	ARDUINO NANO	VARIOS		ARD-NANOCH
14	1	PCB ELECTRONICA	VARIOS		PCB-ARDUINO
13	1	AMPLIFICADOR DE SEÑAL	VARIOS		FC-37
12	1	SENSOR DE CORRIENTE	VARIOS		ACS712-05
11	1	ZUMBADOR	VARIOS		GROOVE BUZZER
10	1	ANTENA 915MHz	VARIOS		WRL-14876
9	1	PLANCHA DE ACERO 2MM		AISI304	
8	1	BASE DE MICROFONO	VARIOS		
7	1	CAJA RITIAL		AISI1010	EB 1546.500
6	1	FUENTE DE 5V-1,2A	VARIOS		PSK-56C-5-T
5	1	BATERIA DE LITIO 3.7V 1200mAh	VARIOS		803048
4	1	PCB POTENCIA	VARIOS		PCB-UPS
3	1	PROTECTOR DE BATERIA	VARIOS		DD05CY5A
2	1	UPS DE 5V	VARIOS		DD05CY5B
1	1	DETECTOR DE LLUVIA	VARIOS		FC-37
POS. CANT.		DESCRIPCION	NORMA	MATERIAL	OBSERVACIONES

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ					
FACULTAD DE CIENCIAS E INGENIERÍA - ESPECIALIDAD: ING. MECATRÓNICA					
METODO DE PROYECCIÓN	TRABAJO DE FIN DE CARRERA 2 - 10M2		ESCALA		
	ENSAMBLE NODO SENSOR		1:1		
20142577	MURATA ARCA, RENZO MITSUO		FECHA:		2019.06.29
			LAMINA:		A1