

**UNIVERSIDAD POLITÉCNICA SALESIANA  
SEDE QUITO**

**CARRERA:  
INGENIERÍA ELECTRÓNICA**

**Trabajo de titulación previo a la obtención del título de:  
INGENIEROS ELECTRÓNICOS**

**TEMA:  
DISEÑO DE UNA RED WSN PARA EL EDIFICIO SAINT CHARBEL  
GESTIONADO A TRAVÉS DE UNA PLATAFORMA IoT DE CÓDIGO  
ABIERTO**

**AUTORES:  
ANDRÉS JEFFERSON CHINACHI QUISHPE  
JORGE ALEXANDER RIOS REVELO**

**TUTOR:  
JHONNY JAVIER BARRERA JARAMILLO**

**Quito, Octubre del 2020**

## CESIÓN DE DERECHOS DE AUTOR

Nosotros Andrés Jefferson Chinachi Quishpe y Jorge Alexander Rios Revelo con documentos de identificación No 1725278384 y No 1724179377 respectivamente, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación intitulado: DISEÑO DE UNA RED WSN PARA EL EDIFICIO SAINT CHARBEL GESTIONADO A TRAVÉS DE UNA PLATAFORMA IOT DE CÓDIGO ABIERTO, mismo que ha sido desarrollado para optar por el título de: Ingenieros Electrónicos, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.



.....

Andrés Jefferson Chinachi Quishpe  
Cédula: 1725278384



.....

Jorge Alexander Rios Revelo  
Cédula: 1724179377

Quito, Octubre del 2020.

## **DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR**

Yo declaro que bajo mi dirección y asesoría fue desarrollado el Proyecto Técnico, DISEÑO DE UNA RED WSN PARA EL EDIFICIO SAINT CHARBEL GESTIONADO A TRAVÉS DE UNA PLATAFORMA IOT DE CÓDIGO ABIERTO realizado por Jorge Alexander Rios Revelo y Andrés Jefferson Chinachi Quishpe, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerado como trabajo final de titulación.

Quito, Octubre del 2020



Ing. Jhonny Javier Barrera Jaramillo

Cédula de identidad: 1400378475

## **DEDICATORIA**

Dedico este Trabajo de Titulación con gran cariño, a mi padre Jorge Rios y a mi madre Carmita Revelo por ser el motor de mi vida, por todo el esfuerzo que realizan para que pueda culminar mi formación profesional, por todo el apoyo, cariño y por ser mis ejemplos a seguir en la vida, gracias por todo papitos. A mi hermanita que es el regalo más grande que me ha dado la vida. A mi tío Fausto Revelo quien es uno de los pilares fundamentales en mi vida, por ser el ejemplo y apoyo, gracias por tu cariño y consejos ñañito. A todas las personas que marcaron mi desarrollo personal y profesional.

*Jorge Alexander Rios Revelo*

El presente proyecto dedico a mis queridos y amados padres, hermanos y mi sobrinita por ayudarme a ser quien soy y apoyarme en todos los sentidos a lo largo de mi vida, por dar sentido a mi vida y mostrarme los caminos a seguir, siempre estar a mi lado y darme fuerza en mis momentos de cansancio, por toda la felicidad y amor compartido todo este tiempo, son mis recuerdos e historias, son mi fortaleza y lo mejor de mi vida.

*Andrés Jefferson Chinachi Quishpe*

## **AGRADECIMIENTO**

Primeramente, agradezco a Dios por darme salud, guía y derramar bendiciones sobre cada uno de los miembros de mi familia. A mis padres y hermana, tíos, primos y abuelitos por su apoyo incondicional, consejos, enseñanzas, cariño y compañía. A mis amigos Marce, Moni, Yess, Liz, Andrés, Edwin, Fausto, Bryan, Christopher, Xavier y demás amigos, por haber compartido junto a mis las diferentes etapas de la carrera universitaria y los mejores momentos en mi carrera, por apoyarme y porque a más de ser amigos se han convertido en familia. A la Universidad Politécnica Salesiana y a los diferentes ingenieros que la conforman, por sus enseñanzas, guía y conocimientos impartidos a lo largo de mi formación profesional. Agradezco al Ingeniero Jhonny Barrera por su guía, apoyo, paciencia, conocimientos impartidos y sobre todo porque a más de ser mi profesor es mi amigo, ya que ha sido parte fundamental para la culminación exitosa del presente proyecto. Todo mi agradecimiento por su tiempo, comprensión y sus grandes consejos que los llevaré siempre en el corazón.

*Jorge Alexander Rios Revelo*

Agradezco a mi madre Olga Quishpe y a mi padre Luis Chinachi por darme todo el amor y apoyo en todos los años de vida, por incentivar me a mejorar cada día y por siempre preocuparse de mí, por enseñarme tanto todo este tiempo, enseñarme que todo vale la pena, que nunca es demasiado tarde para hacer lo que quieres, que de todas las cosas siempre hay algo bueno, por ser guías y ejemplos. A mis queridos hermanos Christian, Alex, Tania y mi sobrinita Ivette por ayudarme en todo siempre y brindarme sus consejos, ayudarme a superarme tanto física como mentalmente, por todos los momentos compartidos y ser ejemplos de lucha, por sorprenderme siempre con sus gestos y por ser siempre un gran apoyo. Ustedes son mi mejor ejemplo y siempre estarán en mis acciones y pensamientos.

*Andrés Jefferson Chinachi Quishpe*

## ÍNDICE GENERAL

<b>CESIÓN DE DERECHOS DE AUTOR .....</b>	<b>I</b>
<b>DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR .....</b>	<b>II</b>
<b>DEDICATORIA.....</b>	<b>III</b>
<b>AGRADECIMIENTO .....</b>	<b>IV</b>
<b>RESUMEN.....</b>	<b>XII</b>
<b>ABSTRACT .....</b>	<b>XIII</b>
<b>INTRODUCCIÓN .....</b>	<b>XIV</b>
<b>CAPÍTULO 1.....</b>	<b>1</b>
<b>1. ANTECEDENTES .....</b>	<b>1</b>
1.1. Planteamiento del Problema.....	1
1.2. Justificación .....	1
1.3. Objetivos .....	2
1.3.1 Objetivo General .....	2
1.3.2 Objetivos Específicos.....	2
1.4. Marco Conceptual .....	3
1.4.1. Redes de Sensores Inalámbricos o WSN .....	3
1.4.1.1. Elementos de la red WSN .....	3
1.4.1.2. Topologías de redes WSN.....	5
1.4.2. Internet de las Cosas o IoT.....	5
1.4.3. Protocolos de Comunicación Inalámbrica .....	7
1.4.4. Plataforma IoT .....	8
1.4.5. Cloud.....	8
<b>CAPÍTULO 2.....</b>	<b>10</b>
<b>2. ESTUDIO INICIAL.....</b>	<b>10</b>
2.1. Descripción de la Edificación .....	10
2.1.1. Ubicación .....	10
2.1.2. Misión .....	10
2.1.3. Visión .....	10

2.1.4. Descripción de la Infraestructura de la Edificación .....	10
2.2. Descripción de la Red actual.....	11
2.2.1. Arquitectura .....	11
2.2.2. Topología Física.....	12
2.2.3. Topología Lógica .....	12
2.2.4. Equipamiento Activo .....	13
2.2.5. Equipamiento Pasivo.....	13
2.3. Problemas Detectados .....	13
2.4. Requerimientos .....	14
<b>CAPÍTULO 3.....</b>	<b>16</b>
<b>3. DISEÑO DE LA RED IoT.....</b>	<b>16</b>
3.1. Generalidades .....	16
3.2. Metodología .....	16
3.3. Diseño Físico de la red WLAN .....	16
3.3.1. Topología Física de la red WLAN.....	17
3.3.2. Distribución de los APs.....	17
3.3.3. Ubicación de Sensores .....	18
3.3.4. Distribución de Sensores.....	19
3.4. Diseño Lógico de la Red WLAN .....	22
3.4.1. Arquitectura de Red .....	22
3.4.2. Topología Lógica de la red WLAN .....	22
3.4.3. Parámetros de configuración.....	23
3.4.3.1. VLAN's.....	23
3.4.3.2. Planeación de direcciones Lógicas.....	24
3.4.3.3. Mapas de Calor .....	25
3.4.4. Selección de equipamiento para la WLAN.....	26
3.4.5. Cálculo del ancho de banda.....	26
3.5. Diseño de la Red WSN .....	29
3.5.1 Elección de Protocolo de Comunicación Inalámbrica .....	29
3.5.2. Equipamiento para la WSN.....	29
3.5.2.1. Elección de Motes o Nodos .....	29

3.5.2.2. Fuentes de Alimentación.....	30
3.5.2.3. Elección de Sensores.....	30
3.6. Componentes IoT .....	32
3.6.1. Elección de Plataforma IoT.....	32
3.6.2. Elección de Protocolo de intercambio de mensajes .....	33
3.6.3. Elección de Base de Datos .....	34
3.6.4. Elección de Servidores .....	34
3.7. Simulaciones .....	34
3.7.1. Configuración de los dispositivos .....	35
3.7.2. Resultados .....	35
3.8. Análisis Económico .....	37
3.8.1. Costos de diseño de hardware .....	37
3.8.2. Costos de diseño de software .....	37
3.8.3. Costos de implementación .....	38
3.8.4. Gastos de operación .....	39
3.8.5. Valor actual neto (VAN).....	39
3.8.6. Tasa interna de retorno (TIR).....	40
3.8.7. Periodo de recuperación de inversión (PRI) .....	40
3.8.8. Relación beneficio/costo .....	41
3.8.9. Análisis.....	42
<b>CAPÍTULO 4.....</b>	<b>43</b>
<b>4. SIMULACIÓN Y PRUEBAS.....</b>	<b>43</b>
4.1. Generalidades .....	43
4.2. Configuración.....	44
4.2.1. Configuración de red WSN.....	44
4.2.2. Simulación y configuración de servidores .....	44
4.2.2.1. Protocolo para el intercambio de mensajes.....	44
4.2.2.2. Base de Datos .....	45
4.2.2.3. Plataforma IoT .....	45
4.2.2.4. Integración Cloud.....	51
4.3. Pruebas y Resultados .....	51



4.3.1. Funcionamiento de la Plataforma IoT .....	51
4.3.2. Lectura de Sensores .....	52
<b>CONCLUSIONES.....</b>	<b>58</b>
<b>RECOMENDACIONES.....</b>	<b>59</b>
<b>BIBLIOGRAFÍA.....</b>	<b>60</b>
<b>ANEXOS .....</b>	<b>63</b>

## ÍNDICE DE FIGURAS

Figura 1.1 Elementos de una red WSN .....	3
Figura 1.2 Componentes de un nodo sensor .....	4
Figura 1.3 Arquitectura IoT .....	6
Figura 2.1 Ubicación Referencial .....	10
Figura 2.2 Arquitectura de Red.....	11
Figura 3.1 APs Planta Baja .....	18
Figura 3.2 Distribución Planta Baja .....	20
Figura 3.3 Topología lógica de la Edificación .....	23
Figura 3.4 Mapa de intensidad de señal de la Planta Baja .....	26
Figura 3.5 NodeMCU .....	29
Figura 3.6 Sensor de humo y gas butano .....	30
Figura 3.7 Sensor de temperatura y humedad .....	31
Figura 3.8 Sensor de fuego .....	31
Figura 3.9 Sensor de nivel.....	31
Figura 3.10 Diagrama de la simulación para la WLAN .....	34
Figura 3.11 Dirección del AP8 correspondiente al piso 6 .....	35
Figura 3.12 Dirección obtenida mediante DHCP para la laptop 2.....	36
Figura 3.13 Dirección obtenida mediante DHCP para la laptop 1.....	36
Figura 3.14 Dirección obtenida mediante DHCP para el Smartphone 2 .....	37
Figura 4.1 Configuración sensor de nivel .....	44
Figura 4.2 Interfaz web openHAB .....	46
Figura 4.3 Archivos de configuración.....	46
Figura 4.4 Things en openHAB .....	47
Figura 4.5 Archivo de configuración para Things .....	47
Figura 4.6 Items en openHAB .....	48
Figura 4.7 Archivo de configuración para Items .....	48
Figura 4.8 Archivo de configuración para Sitemap .....	49
Figura 4.9 Interfaz Sitemap.....	49
Figura 4.10 Archivo de configuración para Rules .....	49
Figura 4.11 Base de datos openhab_db.....	50
Figura 4.12 Maqueta realizada.....	52
Figura 4.13 Prototipo de cisterna .....	52

Figura 4.14 Notificaciones en Telegram.....	57
Figura 4.15 Notificaciones en Whatsapp .....	57

## ÍNDICE DE TABLAS

Tabla 2.1 Distribución de la Edificación.....	11
Tabla 3.1 Distribución de APs .....	17
Tabla 3.2 Distribución de Sensores en Planta Baja.....	19
Tabla 3.3 Distribución de Sensores en Subsuelos.....	20
Tabla 3.4 Distribución de Sensores en Pisos 1 y 2 .....	21
Tabla 3.5 Distribución de Sensores en Pisos del 3 al 11.....	21
Tabla 3.6 Distribución de Sensores en Piso 12 y Terraza.....	21
Tabla 3.7 Número total de Sensores en el Edificio .....	22
Tabla 3.8 Planificación de crecimiento .....	24
Tabla 3.9 Direccionamiento de las vlan's del edificio Saint Charbel .....	24
Tabla 3.10 Distribución de SSID .....	25
Tabla 3.11 Direcciones de los equipos de red.....	25
Tabla 3.12 Tamaño de paquetes .....	27
Tabla 3.13 Capacidad de Ancho de Banda .....	28
Tabla 3.14 Selección de Plataforma IoT de código abierto .....	32
Tabla 3.15 Costo diseño de hardware .....	37
Tabla 3.16 Costo diseño de software .....	37
Tabla 3.17 Resumen del costo de implementación .....	38
Tabla 3.18 Gastos de operación .....	39
Tabla 3.19 Cálculo de VAN.....	40
Tabla 3.20 Cálculo de TIR.....	40
Tabla 3.21 Cálculo de PRI .....	41
Tabla 3.22 Relación beneficio/costo .....	41
Tabla 3.23 Tabla resumen .....	42
Tabla 4.1 Resultados sector Lavandería.....	54
Tabla 4.2 Resultados sector Gimnasio .....	55
Tabla 4.3 Resultados Cisterna.....	55
Tabla 4.4 Resultados sector Conserje .....	56
Tabla 4.5 Resultados sector Salón Comunal.....	56

## **RESUMEN**

La integración de tecnologías de sensores inalámbricos (WSN) e IoT (Internet de las cosas), comprende sin duda una innovadora propuesta que en los últimos años ha ido fortaleciendo los procesos de automatización de diferentes escenarios de la medicina, agricultura, industria, edificios y ciudades inteligentes. Cada vez es más frecuente observar soluciones tecnológicas que permiten el monitoreo y la gestión de eventos en tiempo real mediante el envío de información usando Internet. El presente proyecto establece el diseño de una red WSN que trabaja conjuntamente con la tecnología IoT desplegados sobre una red WLAN, cuyos nodos usan el protocolo Wi-Fi con el fin de mejorar la eficiencia en el uso de recursos y servicios de una edificación habitacional y que es gestionada mediante la plataforma libre OpenHAB.

Para la simulación y las pruebas de la red IoT se utilizó la placa NodeMCU, mientras que para la red WLAN se consideraron equipos de la marca Cisco, los sensores considerados en la red WSN se encargan de detección de temperatura, de humo, de gas GLP, de nivel de agua y de llama.

La simulación de la red WSN integrada con IoT se realizó mediante una maqueta que representa las instalaciones de uno de los ambientes del edificio y su respectiva gestión usando la plataforma OpenHAB como un servidor concentrador de datos para enviarlos a la nube para la gestión y monitoreo de los sensores usando notificaciones en tiempo real.

## **ABSTRACT**

The integration of wireless sensor technologies (WSN) and IoT (Internet of Things) comprises an innovative proposal that in recent years has been strengthening the automation processes of different scenarios of medicine, agriculture, industry, buildings and smart cities. It is increasingly common to observe technological solutions that allow monitoring and management of events in real time by sending information using the Internet. This project establishes the design of a WSN network that works in conjunction with IoT technology deployed over a WLAN network, whose nodes use the Wi-Fi protocol in order to improve the efficiency in the use of resources and services of a housing building and which is managed by the free OpenHAB platform.

For IoT network testing and simulation, the NodeMCU board was used, while for the WLAN network Cisco-branded equipment was considered, the sensors considered in the WSN network are responsible for temperature, smoke, LPG gas, water level, and flame detection.

The simulation of the WSN network integrated with IoT was performed by means of a model that represents the installations of one of the building environments and their respective management using the OpenHAB platform as a data hub server to send them to the cloud for the management and monitoring of sensors using real-time notifications.

## INTRODUCCIÓN

El diseño del presente proyecto técnico está dividido en cuatro capítulos, a lo largo de los cuales se documenta el diseño, simulación y pruebas de una Red WSN gestionada a través de una plataforma IoT de código abierto para el Edificio Saint Charbel, perteneciente a la constructora Saud. En el primer capítulo, se detallan los antecedentes del proyecto y la recopilación de la información necesaria para establecer el fundamento teórico que respalda el diseño de una red WSN y su integración a una red IoT.

En el segundo capítulo, se establecen los requerimientos de automatización del edificio, realizando el estudio del diseño estructural para realizar una propuesta de diseño de red WSN, permitiendo la automatización de sistemas que prioricen el confort y precautelen la seguridad de sus habitantes ante problemas de servicios.

En el tercer capítulo, se plantea una solución que incluye una red WLAN que permita la comunicación de los sensores y actuadores de la red WSN, basado en un enfoque PPDIOO, para lo cual se desarrolla un diagrama físico y lógico de distribución y ubicación según los planos estructurales proporcionados por la constructora, se realiza un estudio de factibilidad económica en la que se evalúa todos los costos que involucran el diseño de la red WLAN, WSN y la plataforma de gestión con el respectivo análisis y selección de los fabricantes de sensores y actuadores existentes en el mercado, además se realiza una comparación de las diversas plataformas de código abierto para la gestión de los dispositivos IoT que permita interoperabilidad, gestión remota y monitoreo en tiempo real.

Finalmente, en el capítulo cuarto, se pone a prueba la operación de la plataforma IoT y su funcionamiento con diferentes sensores. Para la demostración de la funcionalidad de la propuesta se realiza un prototipo que consiste en una maqueta de la planta baja del edificio Saint Charbel, en el cual se visualiza la implementación de la red WSN y su respectiva gestión desde un servidor IoT de acuerdo al análisis de requerimientos.

# CAPÍTULO 1

## 1. ANTECEDENTES

### 1.1. Planteamiento del Problema

El edificio Saint Charbel es un proyecto ambicioso de la empresa SAUD, que abarca la construcción de una torre moderna de 12 pisos donde se ubicarán 96 departamentos, tiendas comerciales y otros ambientes de convivencia comunitaria (gimnasio, salón de juegos, salón de uso múltiple, etc.).

Como parte de su propuesta innovadora, los proyectistas del edificio han decidido incluir en su diseño y planimetría el uso de una red de Sensores Inalámbricos (WSN) y la aplicación del Internet de las cosas (IoT) para la gestión, control y seguridad de los sistemas automatizados que hagan más confortable y segura la vida de los futuros residentes. Esta red utilizará dispositivos tecnológicos que realicen lecturas continuas de condiciones energético-ambientales tales como un sistema de climatización de áreas, control de sensores de humo, lecturas de llama, detectores de gas butano, control de llenado de tanques y cisternas de agua.

Todo se realizará sobre una red de datos WLAN que permita la comunicación de los diferentes sensores y que además permita aprovechar la flexibilidad e interoperabilidad de una herramienta Open Source de código abierto para la gestión de la información desde la nube.

### 1.2. Justificación

El diseño y construcción de una edificación destinada a vivienda o comercio ha evolucionado y pasado por diferentes etapas a lo largo de la historia, pero siempre ha tenido como objetivo el confort de sus habitantes. Actualmente, las empresas constructoras procuran innovar en cuanto a sus infraestructuras modernas que ofrecen incontables servicios entre los cuales se destacan la seguridad basada en controles de acceso utilizando tecnología biométrica, control ante incendios, sistemas novedosos para controlar la distribución de recursos de agua, de gas, detección de fugas, entre otros.



La implementación de estas tecnologías requiere la integración de sensores y actuadores ubicados en sitios específicos que permiten obtener y enviar lecturas en tiempos constantes hacia un punto central de control. Sin embargo, este tipo de soluciones representan un costo elevado debido a que en muchos casos deben sujetarse a la contratación de un proveedor específico y/o al uso de dispositivos de fabricantes con interoperabilidad limitada. Esta situación sin duda complica la gestión y detección de fallas o la aplicación de estrategias de mantenimiento preventivo.

En este sentido, el diseño de una red WSN en un edificio de viviendas caracterizado por brindar un alto índice de integración de sensores y actuadores multifabricantes y cuya gestión se realice a través de una plataforma IoT Open Source, representa sin duda un atractivo para una empresa constructora y marca una alternativa de innovación para que diversas organizaciones adopten este tipo de tecnología.

### **1.3. Objetivos**

#### **1.3.1 Objetivo General**

Diseñar la red WSN para automatizar el edificio Saint Charbel de la constructora SAUD utilizando una plataforma IoT de código abierto para su gestión desde la nube.

#### **1.3.2 Objetivos Específicos**

- Establecer los requerimientos de automatización del edificio Saint Charbel a partir del estudio de su diseño estructural.
- Realizar un estudio comparativo de las diversas plataformas IoT de código abierto y su compatibilidad con las tecnologías de sensores y actuadores existentes en el mercado.
- Diseñar la red WSN para la conexión de los sensores y actuadores sobre la red LAN y WLAN desde los diferentes niveles del edificio con el servidor IoT.
- Desarrollar un prototipo para demostrar la funcionalidad de la red WSN y su gestión en la nube a través de la plataforma IoT seleccionada.
- Analizar la factibilidad económica para la implementación de la red WSN propuesta.

## 1.4. Marco Conceptual

### 1.4.1. Redes de Sensores Inalámbricos o WSN

Las redes de sensores inalámbricas o WSN por sus siglas en inglés Wireless Sensors Network, son un tipo de red la cual está formada por diferentes tipos de dispositivos sensores que permiten la medición de variables físicas o eventos en un lugar específico y utilizan ondas electromagnéticas para su comunicación, para así recopilar los datos del lugar, procesarlos y enviarlos a un nodo central.

Estas redes están compuestas de nodos los cuales tienen en su interior dos unidades, una de control y otra de medición. Los nodos sensores envían cada determinado tiempo a un nodo central que es el encargado de recopilar los datos de todos los sensores que están trabajando, así como también enviar datos requeridos por los diferentes sensores.

Estas redes en la actualidad son muy utilizadas debido a que su implementación en diferentes sitios es de manera sencilla, además de que por la fácil configuración de los diversos elementos pueden ser nodos emisores, receptores o brindar otros servicios que componen la red WSN. La norma IEEE 802.15.4 presenta características que definen a este tipo de redes especialmente para las capas: física y de control de acceso al medio, a la vez estos equipos pueden ser de funciones completas o reducidas.

#### 1.4.1.1. Elementos de la red WSN

Una red WSN está compuesta por los elementos mostrados en la figura 1.1.

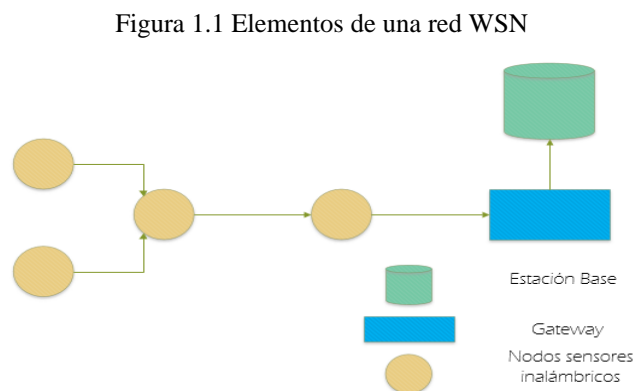


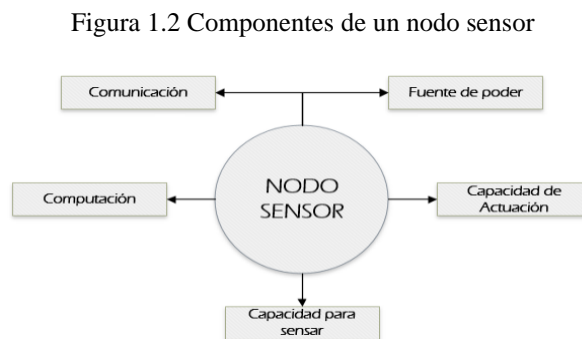
Diagrama representativo de elementos de una red WSN. Elaborado por: Chinachi Andrés, Rios Jorge.

- Estación Base: Tiene la función de recolectar, organizar y sincronizar toda la información proveniente de todos los diversos nodos. Esta estación puede ser

un computador o un hardware dedicado a esta función, a su vez tiene la capacidad de comunicarse con un servidor web para aplicaciones que en conjunto funcionen con la red.

- Nodos sensores: son dispositivos de pequeño tamaño que trabajan generalmente usando baterías de voltaje reducido. Estos nodos tienen diferentes sensores, que a su vez se componen de elementos que se encuentran conectados entre sí, los cuales son: motes o nodos y sensores.
- Motes o Nodos: Elemento que se encarga del almacenamiento, comunicación, evaluación, cálculo y operación, estos pueden ser nodos finales o intermedios
- Sensores: Elementos que tienen la capacidad de medir diferentes fenómenos físicos, como son temperatura, luz, vibración, entre otros. Esta información es obtenida y transformada a señales eléctricas.

Un nodo sensor debe comprender un tipo de comunicación, una capacidad para censar, capacidad de actuación, capacidad de cómputo y una fuente de poder para su funcionamiento, el diagrama de componentes es mostrado en la figura 1.2



Representación de los componentes de un sensor. Elaborado por: Chinachi Andrés, Rios Jorge.

Para cumplir con estas funciones el nodo sensor está constituido por un procesador, una memoria de almacenamiento, un transceptor, un conversor análogo a digital ADC y una fuente de poder.

- Gateway: elemento encargado del enlace entre la red WSN y el computador o la red TCP/IP que se encuentre en el lugar.

#### **1.4.1.2. Topologías de redes WSN**

Las topologías que puede usar la WSN depende del diseño y de las tareas que debe cumplir en el lugar en el que se desea instalar, teniendo las siguientes posibilidades:

- Topología en estrella o de un solo salto: En esta los nodos sensores se comunican directamente con la estación base, reduciendo así las posibilidades de fallo y concentrando los problemas de cada nodo por separando.
- Topología árbol o clúster: En esta topología cada uno de los nodos se conecta a otro nodo que tiene más rango debido a sus capacidades, estableciendo un enlace entre el nodo final a la estación base mediante nodos intermediarios.
- Topología en malla: Es descentralizada ya que cada nodo tiene la posibilidad de comunicarse con otros nodos para el envío de la información, teniendo varios caminos para llegar a la estación base, evitando así fallas en el caso de que algún nodo falle.

#### **1.4.2. Internet de las Cosas o IoT**

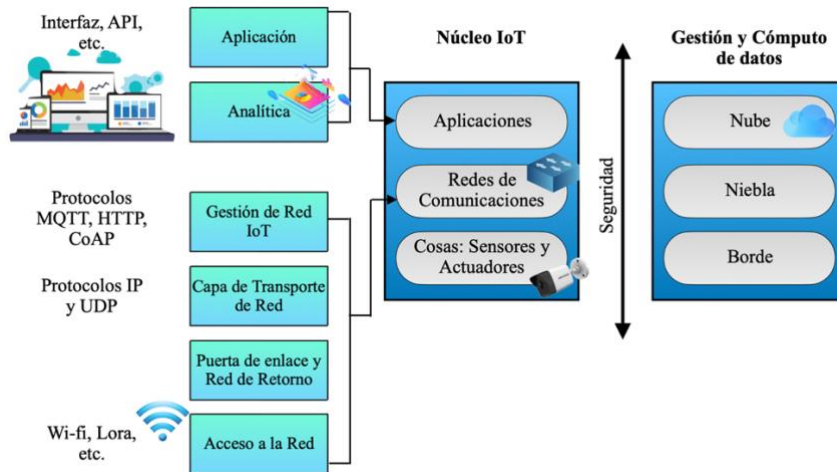
Es una tecnología que permite la conexión de cualquier dispositivo del mundo físico hacia internet, dichos dispositivos comúnmente son denominados como sensores, los cuales recopilan información y la envían hacia un controlador o dispositivo de cómputo con conexión a internet, permitiendo una interacción y comunicación con usuarios, todo esto a través de un medio de comunicación inalámbrico.

Los sensores son utilizados para la recopilación de información, sin embargo, el permitir el análisis y cómputo de dicha información debe estar enfocado en la presentación de un servicio o de una acción por medio de actuadores, logrando incursionar en aplicaciones que buscan mejorar la calidad de vida de las personas y facilitar su trabajo; la capacidad de envío y recopilación de información convierte a un dispositivo en inteligente y sus aplicaciones más comunes se encuentran en industrias inteligentes, edificios inteligentes, ciudades inteligentes, medicina, entre otros.

IoT posee una arquitectura basada en capas, las cuales se presentan en la figura 1.1, estas capas buscan relacionar los componentes de IoT e indicar cómo se garantiza seguridad, escalabilidad e interoperabilidad en la información que se maneja, además,

de presentar un modelo de operación basado en Núcleo IoT y gestión y cómputo de datos.

Figura 1.3 Arquitectura IoT



Representación de arquitectura IoT. Elaborado por: Chinachi Andrés, Rios Jorge.

La capa cosas corresponde a los sensores que recopilan la información y los actuadores que realizan una acción en consecuencia a la lectura del sensor, se encuentra en el borde de una red IoT representando un contacto directo o intervención directa con la variable que se desea medir.

La capa redes de comunicaciones comprende la conexión de dispositivos como gateways o controladores que reciben la información de los sensores y permiten su salida a internet, utilizando tecnologías inalámbricas como Wi-Fi que es la común o estándares de comunicación propios de la tecnología IoT como son Lora o Zigbee, además comprende protocolos tanto para el transporte de datos (IP, UDP), así como protocolos para gestión de mensajes (MQTT, HTTP, CoAP) que implican una comunicación segura entre dispositivos y su salida a internet, generalmente esta capa se encuentra en la denominada niebla, donde se tiene un dispositivo central (gateway o controlador) que recibe toda la información recopilada en los bordes de la Red para llevarlos a internet.

Finalmente, en la capa aplicaciones se encuentra el software de plataforma IoT que permite almacenamiento, análisis y procesamiento de la información, todo esto

orientado al ofrecimiento de servicios además de permitir una integración con servicios en la nube y API's que permiten la toma de decisiones de forma inteligente.

### **1.4.3. Protocolos de Comunicación Inalámbrica**

En una red IoT, los dispositivos necesitan un medio de comunicación para transmisión y recepción de información, generalmente se utiliza un medio inalámbrico, para lo cual se han desarrollado diferentes protocolos de comunicación de los cuales los más comunes son:

- **RFID:** conocido como Identificación por Radio Frecuencia, basa su operación en la asociación e identificación de dispositivos mediante tag's o etiquetas, puede ser utilizada de forma pasiva sin la incorporación de baterías y con la función de etiquetar algún artículo, de forma activa con la incorporación de sensores que permiten monitorear distintas variables eléctricas y posee sus propios recursos necesarios para su operación, finalmente pueden ser semi-pasivas sirviendo como identificadores con la capacidad de guardar información única para mejorar la seguridad en el proceso de autenticación.
- **Bluetooth:** es una de las tecnologías más utilizadas hoy en día y posee diferentes versiones siendo su última actualización la versión 5.0, tiene un bajo consumo de energía lo que la hace ideal para ser utilizada en aplicaciones de monitoreo de salud, sin embargo, posee un corto alcance de comunicación y su velocidad de transmisión varía dependiendo de la versión.
- **Zigbee:** es una tecnología centrada en el bajo consumo de energía, bajo costo y corto alcance con baja transmisión de datos, es mayormente utilizada para dispositivos en ambientes residenciales y busca cuidar la vida de la batería de los dispositivos.
- **802.11:** es uno de los protocolos mayormente utilizado en redes IoT debido a que corresponde a la tecnología Wi-Fi desplegada a nivel mundial y en la mayoría de los hogares, se han desarrollado diferentes versiones de las cuales se ha buscado siempre mejorar la velocidad de transmisión, seguridad en la información y rango de alcance, su rango de frecuencia para operación se ha especificado en 2.4 y 5 GHz.

- LoRaWAN: Es una tecnología desarrollada para largas distancias ofreciendo servicios IoT en lugares de difícil acceso, ciudades inteligentes y cualquier aplicación que requiera comunicación de dispositivos basados en nodos finales que se comunican con un gateway central, su largo alcance depende del rango de frecuencias que se vaya utilizar generalmente puede ir de los 423 a los 928 MHz y su velocidad de transmisión y recepción depende de diversos factores como distancia, línea de vista y potencia de transmisión.

#### **1.4.4. Plataforma IoT**

Una plataforma se define como un software especializado en la creación de aplicaciones soluciones y servicios Cloud, presentan características como compatibilidad, integraciones con diferentes servicios, seguridad, costo y flexibilidad.

Dentro de las consideraciones de una Red IoT, se encuentra la implementación de una plataforma que pueda servir como punto central de gestión de información, desarrollo y ejecución de aplicaciones, procesamiento y accionamiento remoto de actuadores, además de servir como repositorio con posibilidad de análisis de datos con operación en tiempo real.

#### **1.4.5. Cloud**

La nube es una plataforma basada en servicios de internet que permite almacenamiento de datos, seguridad, y acceso a servicios virtuales a bajo costo, velocidad y productividad. Es un gran centro de datos que busca ofrecer servicios externos brindando al usuario la posibilidad de independencia en cuanto locación y permitiendo el ofrecimiento de servicios que se clasifican en IaaS, SaaS y PaaS.

- IaaS: Infraestructura como Servicio, por medio de una suscripción, proporciona a sus clientes recursos que incluyen servicios de almacenamiento y máquinas virtuales con alta escalabilidad necesarios para satisfacer el requerimiento del usuario, además de permitir realizar una configuración independiente.
- SaaS: Software como Servicio, el proveedor proporciona el software o las aplicaciones que sean requeridas por el usuario, las actualizaciones,

mantenimiento y disponibilidad son realizadas únicamente por el proveedor, permitiendo el usuario sólo utilice el servicio.

- PaaS: Plataforma como Servicio, el proveedor proporciona al usuario una interfaz que le permita desarrollar aplicaciones, realizar gestión y herramientas necesarios para un trabajo colaborativo de forma remota, la infraestructura y el software son proporcionadas por el proveedor.

Adicionalmente existen modelos de implementación de cloud, basándose en las necesidades que una organización requiere, los modelos se clasifican en:

- Nube Privada: Es de propiedad de una organización para un uso personalizado únicamente por miembros de la organización.
- Nube Pública: Es ofrecida por proveedores externos los cuales ponen la disponibilidad de recursos y servicios a diferentes organizaciones que realizan el pago por el servicio que se desea utilizar.
- Nube Comunitaria: También conocida como híbrida ya que combina la necesidad de privacidad de administración de datos por una nube privada y la posibilidad de escalabilidad según el servicio utilizado ofrecido por una nube pública, permitiendo una carga equilibrada de datos.



## CAPÍTULO 2

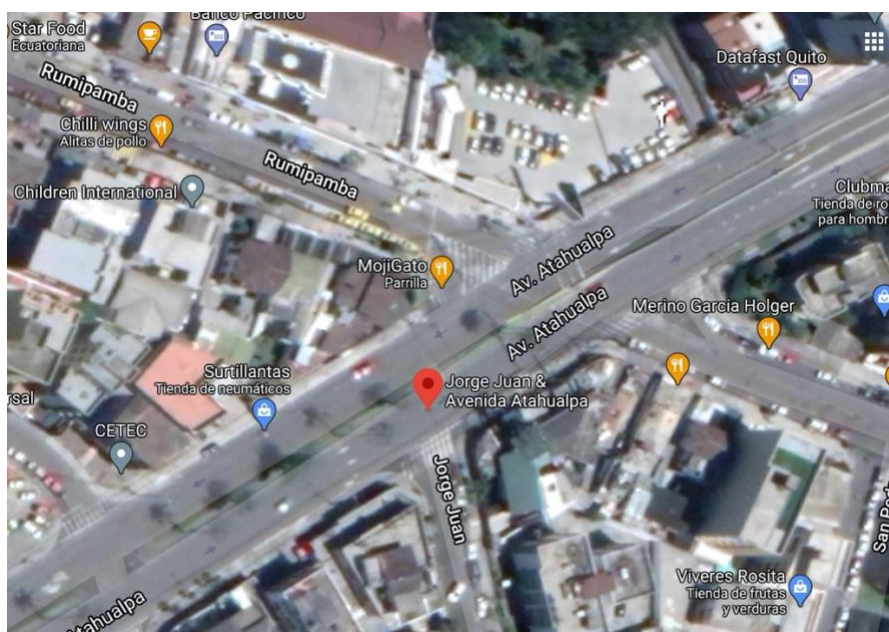
### 2. ESTUDIO INICIAL

#### 2.1. Descripción de la Edificación

##### 2.1.1. Ubicación

La edificación estará ubicada en la ciudad de Quito, en la Avenida Atahualpa y Calle Juan Bayas, su ubicación se muestra en la figura 2.1.

Figura 2.1 Ubicación Referencial



Ubicación referencial del edificio Saint Charbel, Fuente: Google Maps.

##### 2.1.2. Misión

La constructora SAUD tiene como objetivo ser una empresa que prioriza los deseos de sus clientes, mediante su participación en los mismos.

##### 2.1.3. Visión

Su visión es satisfacer las exigencias de sus clientes mediante un trabajo responsable al elaborar proyectos de alta calidad.

##### 2.1.4. Descripción de la Infraestructura de la Edificación

El edificio Saint Charbel consta de 96 departamentos que están distribuidos en 13 plantas y 4 subsuelos, en la tabla 2.1 se muestra la distribución de la edificación.

Tabla 2.1 Distribución de la Edificación

Planta	Distribución
Planta Subsuelo 4	Área de parqueaderos
Planta Subsuelo 3	Área de parqueaderos
Planta Subsuelo 2	Área de parqueaderos
Planta Subsuelo 1	Área de parqueaderos
Planta Baja	Administración, Gimnasio, Guardianía, Patio de juegos, Sala comunal, Sala de juegos, Patio Interno, Vestíbulo, Hidromasaje, Sauna, Turco
Piso 1	4 Departamentos, 3 Locales
Piso 2	7 Departamentos
Piso 3	9 Departamentos
Piso 4	9 Departamentos
Piso 5	9 Departamentos
Piso 6	9 Departamentos
Piso 7	9 Departamentos
Piso 8	9 Departamentos
Piso 9	9 Departamentos
Piso 10	9 Departamentos
Piso 11	9 Departamentos
Piso 12	4 Departamentos
<b>TOTAL</b>	96 Departamentos, 3 Locales, Administración, Gimnasio, Guardianía, Patio de juegos, Sala comunal, Sala de juegos, Patio Interno, Vestíbulo, Hidromasaje, Sauna, Turco

Distribución de espacios en el edificio Saint Charbel. Elaborado por: Chinachi Andrés, Rios Jorge.

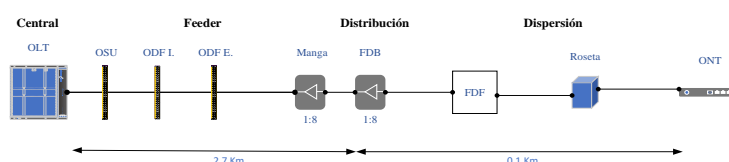
El diseño del edificio incluye un sistema de gas centralizado, una acometida de agua potable, cisternas para reserva de agua y tanques de agua elevados que garanticen el abastecimiento y presión adecuada a todas las instalaciones del edificio, además de un sistema de aire acondicionado centralizado y un sistema de ascensores.

## 2.2. Descripción de la Red actual

### 2.2.1. Arquitectura

La infraestructura de red LAN del edificio Saint Charbel es una red GPON (Gigabit-capable Passive Optical Network) vertical, la cual posee una tipología árbol, usando una arquitectura de tipo FTTH (Fiber To The Home) distribuida, presentada en la figura 2.2.

Figura 2.2 Arquitectura de Red



Arquitectura de red de acceso FTTH con GPON. Elaborado por: Chinachi Andrés, Rios Jorge.

El proveedor de servicio de internet es CNT E.P., el cual utilizó un sistema de distribución con dos niveles de distribución, para el primer nivel, CNT usó una acometida con dos splitters de 1:8 desde el cual se alimenta al FDB (Fiber Demarcation Box) del edificio con 13 fibras ópticas de este, en el FDB se instalan splitters secundarios de 1:8 desde los cuales se alimenta a los diferentes FDF's (Fiber Distribution Frame) a instalarse a lo largo del edificio los cuales son solo de fusión, en donde se instalan cables de fibra óptica tipo drop para la distribución a cada uno de los departamentos.

### **2.2.2. Topología Física**

La red parte de un cuarto de distribución que se encuentra en el subsuelo 1, donde se encuentra la conexión a la acometida de CNT E.P., y del cual parte a dos tableros FDB, y de este a 13 FDFs de paso hacia las rosetas ópticas en cada departamento y en los mismo se tiene un ONT (Optical Network Terminal), de acuerdo a lo indicado la demanda del servicio corresponde a 102 fibras para los abonados, cuya distribución se presenta en el anexo 1. Se usan cuatro fibras ópticas para la acometida conectados a dos niveles de splitters, teniendo así 104 fibras colocadas en toda la instalación de manera distribuida para 102 abonados finales. La topología física donde se evidencia la distribución vertical de la red GPON, se muestra en el anexo 2.

### **2.2.3. Topología Lógica**

La red GPON permite ofrecer un servicio de internet de alta velocidad con la posibilidad de brindar telefonía fija y televisión por cable, todo como un servicio triple play, usando una distribución lógica de las fibras ópticas necesarias para llegar a cada uno de los abonados en la edificación, se presenta en el anexo 3.

La red de datos del edificio posee un plan de direccionamiento que parte de la red 172.20.0.0 con máscara 16, a partir de la cual se realiza subredes para cada uno de los departamentos del edificio a donde se llega con un equipo ONT (Optical Network Terminal), cuyo detalle se presenta en el anexo 4.

#### **2.2.4. Equipamiento Activo**

El equipamiento activo corresponde a un OLT (Optical Line Termination) encargado de control de transmisión de información del proveedor a los usuarios, y los ONT que son puntos terminales de red en el usuario de destino y se instalan en cada departamento.

#### **2.2.5. Equipamiento Pasivo**

El equipamiento pasivo abarca los tableros de distribución FDB y FDFs, que están ubicados en los ductos de telecomunicaciones dentro de cajas metálicas, se tiene rosetas ópticas que es el punto de línea del cliente y donde se conecta la ONT y se instalan en cada departamento, se tiene dos niveles de spliteo con splitter fusionado de 1:8 en la acometida de CNT y en el FDB dando un spliteo final de 1:64.

Para la canalización exterior se tiene una tubería de 110 cm, la cual se encuentra en el suelo del edificio para que por ella pase la canaleta de 20x10cm y de esta llegue a la caja metálica de distribución principal o FDB. Para ello se tiene la construcción de un pozo de revisión de 48 bloques, el cual se encuentra en la parte delantera del edificio.

En la canalización interior se usa canalización vertical con canaleta tipo escalerilla ubicada en los ductos de 20x10 cm. En la canalización horizontal está establecido para los diferentes departamentos la canaleta y tuberías de 10x10cm y 19 mm de diámetro respectivamente, estableciendo un enlace entre la roseta óptica, el ducto y la canaleta vertical.

Para las cajas metálicas del edificio se tiene que la caja de distribución principal FDB es de 100x70x27 cm y las cajas de distribución secundarias FDFs de 40x40x18 cm en el interior del ducto de Telecomunicaciones.

### **2.3. Problemas Detectados**

La red GPON existente en la edificación es independiente y soporta los servicios de telefonía, televisión e internet para cada uno de los departamentos, de manera tal que su gestión depende del usuario y del operador. La administración ha decidido incorporar una red secundaria explícitamente para implementar una red WLAN que

ofrezca una cobertura adecuada en áreas como espacios comunitarios y departamentos de forma tal que permita la instalación de sistemas de seguridad de forma independiente a la red de datos.

La edificación no cuenta con un sistema de monitoreo, prevención y alerta para ambientes con alto índice de riesgo como son fugas de gas esencialmente en cocinas de los departamentos, riesgo de incendios en cuartos de equipos, presencia de humo en espacios comunitarios, en las instalaciones y departamentos, además, no posee un sistema de control adecuado de la cantidad de agua almacenada en los tanques y cisterna de la edificación, ni de temperatura y humedad en zonas propensas a una elevada concentración de calor.

#### **2.4. Requerimientos**

Considerando que no existe una red secundaria, se requiere diseñar una red WLAN que permita brindar una cobertura inalámbrica adecuada en todas las instalaciones de la edificación y que se caracterice por ser administrable y segura.

Se propone la incorporación de sistemas de seguridad y monitoreo basadas en una red WSN de dispositivos sensores y actuadores libres de bajo costo, para lo cual se realiza una selección de sensores para espacios con alto índice de riesgo en la edificación, detallando que los sensores a considerar corresponden a gas GLP doméstico, humedad y temperatura, fuego, humo y ultrasónicos que contemplan las siguientes consideraciones:

- Los sensores de gas GLP doméstico permiten monitorear el aire en espacios determinados y de esta forma detectar la presencia de fugas, generando una alerta para control de la fuga.
- Los sensores de humedad y temperatura permiten realizar lecturas de condiciones ambientales en áreas comunales, para enviar una alerta que permita la activación del sistema de climatización que mantengan un ambiente agradable.
- Los sensores de fuego permiten detectar la presencia de llamas y enviar una alerta para proceder con la revisión de áreas de mayor probabilidad de

combustión como son las cajas de revisión en los diferentes sectores de la edificación.

- Los sensores de humo permiten mediciones de condensación de partículas de humo producido por la combustión, para generar alertas que ayuden a la prevención de incendios.
- El sensor ultrasónico permite realizar un control en el nivel de llenado de la cisterna para la distribución de agua en el edificio dando una alerta cuando el nivel sea bajo o demasiado alto, para de esta forma mantener un nivel adecuado para el bombeo y distribución a las diferentes instalaciones de la edificación.

Se recomienda la incorporación de una plataforma IoT que permita la gestión de los sensores, teniendo en cuenta una interoperabilidad ya que debe ser compatible con cualquier dispositivo existente en el mercado cuya marca sea integrada con la plataforma y que se desee incorporar a la red.

Finalmente, se sugiere la integración de la plataforma IoT con servicios cloud permitiendo una interfaz cuya interacción en la nube sea un servicio como aplicación para uso del administrador y de los empleados de la edificación.

## CAPÍTULO 3

### 3. DISEÑO DE LA RED IoT

#### 3.1. Generalidades

Dado que la edificación se considera de tipo residencial, los requerimientos presentados comprenden la automatización de los sistemas que priorizan el confort y precautelan la seguridad de sus habitantes, por ello, el diseño de la Red WSN considera la elección del protocolo de comunicación más adecuado para interconectar los nodos controladores, sensores, actuadores y demás elementos que correspondan a una óptima solución.

#### 3.2. Metodología

Para el diseño de la red se ha seleccionado un modelo Top down, mismo que permite una perspectiva sistémica de la red para ir satisfaciendo los requerimientos y manteniendo siempre la modularidad en cada una de las etapas que lo comprenden, las cuales son identificación de necesidades, diseño lógico y físico y finalmente pruebas y optimización del diseño, además, se considera un enfoque PPDIOO (Prepare Plan Design Implement Operate Optimize) que permita las acciones necesarias para obtener un buen rendimiento mediante la especificación de las pautas a seguir en cada fase del ciclo de vida de la que se está diseñando, garantizando la disponibilidad e integridad de la red.

#### 3.3. Diseño Físico de la red WLAN

El diseño de la red WLAN se realiza según las recomendaciones del NIST-SP 800-82, en la cual una red OT (Operational Technologies, dedicada a la detección o el cambio en los diferentes procesos físicos que monitorean y controlan) está separada y segmentada tanto de manera lógica como física de la red IT (Information Technology, aplicación de equipos en telecomunicaciones), tomando en cuenta factores como la delimitación de la gestión a una persona o autoridad competente, niveles para la definición de políticas de seguridad, cantidad de tráfico que circulará por la red y las funcionalidades de la red, para establecer una seguridad óptima para el rendimiento de la red limitando accesos e información a los diferentes equipos de la red.

### 3.3.1. Topología Física de la red WLAN

Para la topología física se proporciona una topología en estrella extendida, que permite generar un escenario de cobertura a cada uno de los diferentes pisos de la edificación, implementando un dispositivo central de concentración y conmutadores secundarios que despliegan dispositivos APs (Access Points) encargados de conducir y generar el medio inalámbrico para la comunicación de los sensores. Se utiliza el espacio de administración ubicado en la planta baja, en el cual se encuentra un rack principal de 16 U, con un switch de core con entrada a un ISP, el cual se conecta hacia los diferentes switches de distribución en cada piso. Los diferentes switches de distribución son albergados en racks de 6U que se distribuyen de forma tal que cada uno cubre dos pisos a diferencia del rack central, el cual abarca conexión para el subsuelo 4, subsuelo 3, subsuelo 1, planta baja y piso 1, a partir de cada rack se coloca los APs en los diferentes puntos seleccionados para proporcionar una conectividad adecuada.

### 3.3.2. Distribución de los APs

Los APs se colocaron y distribuyeron en cada piso del edificio, precautelando que no exista obstrucciones y dando cobertura óptima para su uso eficiente, estos estarán colocados en el techo en puntos centrales y estarán distribuidos como se describe en la tabla 3.1.

Tabla 3.1 Distribución de APs

PLANTA	Número de APs	Área a cubrir en m <sup>2</sup>
Planta Subsuelo 4	1	51.10
Planta Subsuelo 3	1	27.06
Planta Subsuelo 1	1	26.90
Planta Baja	5	561.806
Piso 1	6	424.197
Piso 2	7	526.795
Piso 3	9	670.68
Piso 4	9	670.68
Piso 5	9	670.68
Piso 6	9	670.68
Piso 7	9	670.68
Piso 8	9	670.68
Piso 9	9	670.68
Piso 10	9	670.68
Piso 11	9	670.68
Piso 12	5	349.48
Terraza	1	25.485
<b>TOTAL</b>	99	8028.943

Tabla de distribución de APs en la edificación. Elaborado por: Chinachi Andrés, Rios Jorge.





liberarse no se elevan al techo, por eso se colocarán a una distancia de 30 cm del suelo y a una distancia entre 1 y 3 metros de la fuente que se requiera monitorear.

- Los detectores de llama debido a su limitación por su línea de visión a los lugares u objetos que desean vigilar, se incorporan en lugares donde los tiempos de actuación son críticos y donde posibles daños por falsas alarmas es crítico, se toma factores como sensibilidad del detector, distancia máxima a la que detecta, posibles fuentes radiantes, entre otras.
- El sensor ultrasonido se colocará en tanques o reservas de agua para conocer el cambio a lo largo del tiempo de los niveles de agua.
- Los sensores de temperatura se colocan en lugares que posean un sistema de ventilación en cuyas áreas sean comunitarias o susceptibles al aumento de temperaturas.

### 3.3.4. Distribución de Sensores

Para posteriormente segmentar la red WLAN se procede a estimar la cantidad de sensores que se colocan en cada piso del edificio considerando que la Planta Baja es el punto inicial de la red, la distribución de sensores en la planta baja se detalla en la tabla 3.2 y se puede visualizar en la figura 3.2.

Tabla 3.2 Distribución de Sensores en Planta Baja

Descripción lugares Planta Baja	Tipo de Sensor			
	Temperatura	Fuego	Humo	Gas GLP
Administración	-	-	1	-
Vestíbulo	1	1	2	-
Exterior del ascensor	-	-	1	-
Exterior de las escaleras	-	-	1	-
Salón comunal	1	-	4	-
Sala de Juegos	1	1	4	-
Lavandería	-	1	1	-
Cuarto del conserje	-	1	1	1
Cuarto de medidores	-	-	1	-
Acceso a subsuelos	-	-	-	-
Gimnasio	1	-	-	-
Salón de descanso	1	-	1	-
Pasillos de bodega	-	-	1	-
Entrada gimnasio	-	1	-	-
<b>Total, Planta Baja</b>	<b>5</b>	<b>5</b>	<b>18</b>	<b>1</b>

Tabla de distribución de sensores en la planta baja. Elaborado por: Chinachi Andrés, Rios Jorge.

Figura 3.2 Distribución Planta Baja



Distribución de sensores en la Planta Baja. Elaborado por: Chinachi Andrés, Rios Jorge.

En el subsuelo 1,3 y 4 se encuentran sensores para espacios de máquinas y depósitos y su distribución se detalla en la tabla 3.3, en el anexo 6 se presenta la distribución de los sensores en cada uno de los subsuelos.

Tabla 3.3 Distribución de Sensores en Subsuelos

Subsuelo	Descripción	Tipo de Sensor			
		Nivel	Fuego	Humo	Gas
1	Cuarto del Transformador	-	1	1	-
	Cuarto del generador	-	1	1	-
	Tablero de distribución	-	1	-	-
3	Cuarto de bombas	-	1	1	1
4	Cisterna	1	-	1	-
<b>Total, en cada Subsuelo</b>		1	4	4	1

Tabla de distribución de sensores en los subsuelos. Elaborado por: Chinachi Andrés, Rios Jorge.

Los pisos 1 y 2 se diferencian de los demás debido a que cuentan además de los departamentos con locales cuya distribución de sensores se muestra en la tabla 3.4, su lugar de distribución se visualiza en el anexo 6.

Tabla 3.4 Distribución de Sensores en Pisos 1 y 2

Piso	Descripción	Tipo de Sensor		
		Fuego	Humo	Gas GLP
1	Departamentos	4	15	4
	Locales 1 al 3	3	8	-
	Pasillos	-	2	-
2	Suites y Departamento	7	22	7
	Local 3	-	3	-
	Pasillos	-	3	-
<b>Total, Pisos 1-2</b>		14	53	11

Tabla de distribución de los sensores en los pisos 1 y 2. Elaborado por: Chinachi Andrés, Rios Jorge.

Para los pisos 3 al 11 se toma en cuenta que la infraestructura es similar y posee los mismos lugares, por tanto, se considera el piso 3 como referencia, su distribución se indica en la tabla 3.5 y su lugar de distribución se visualiza en el anexo 6.

Tabla 3.5 Distribución de Sensores en Pisos del 3 al 11

	Tipo de Sensor		
	Fuego	Humo	Gas GLP
Suites y departamentos	9	30	9
Pasillos	-	6	-
Total en cada Piso	9	36	9
<b>Total, Piso 3-11</b>	81	324	81

Tabla de distribución de sensores en pisos del 3 al 11. Elaborado por: Chinachi Andrés, Rios Jorge.

El piso 12 se diferencia de los demás debido a que en este se localizan penthouse con mayor área, la distribución de los sensores se indica en la tabla 3.6 y se visualiza en el anexo 6.

Tabla 3.6 Distribución de Sensores en Piso 12 y Terraza

Lugar	Descripción	Tipo de Sensor		
		Fuego	Humo	Gas GLP
Piso 12	Penthouse	4	19	4
	Pasillos	-	3	-
Terraza	Cuarto de calentadores de agua	1	1	1
	Pasillos	-	1	-
<b>Total, Piso 12 y Terraza</b>		5	24	5

Tabla de distribución de los sensores en el piso 12 y terraza. Elaborado por: Chinachi Andrés, Rios Jorge.

El número total de sensores en todo el edificio clasificados por su tipo, se presenta en la tabla 3.7.

Tabla 3.7 Número total de Sensores en el Edificio

<b>Tipo de Sensor</b>	<b>Cantidad</b>
Gas GLP domiciliario	99
Fuego	109
Humo	423
Temperatura	5
Nivel	1
<b>Total Sensores</b>	<b>637</b>

Tabla del número total de sensores. Elaborado por: Chinachi Andrés, Rios Jorge.

### **3.4. Diseño Lógico de la Red WLAN**

#### **3.4.1. Arquitectura de Red**

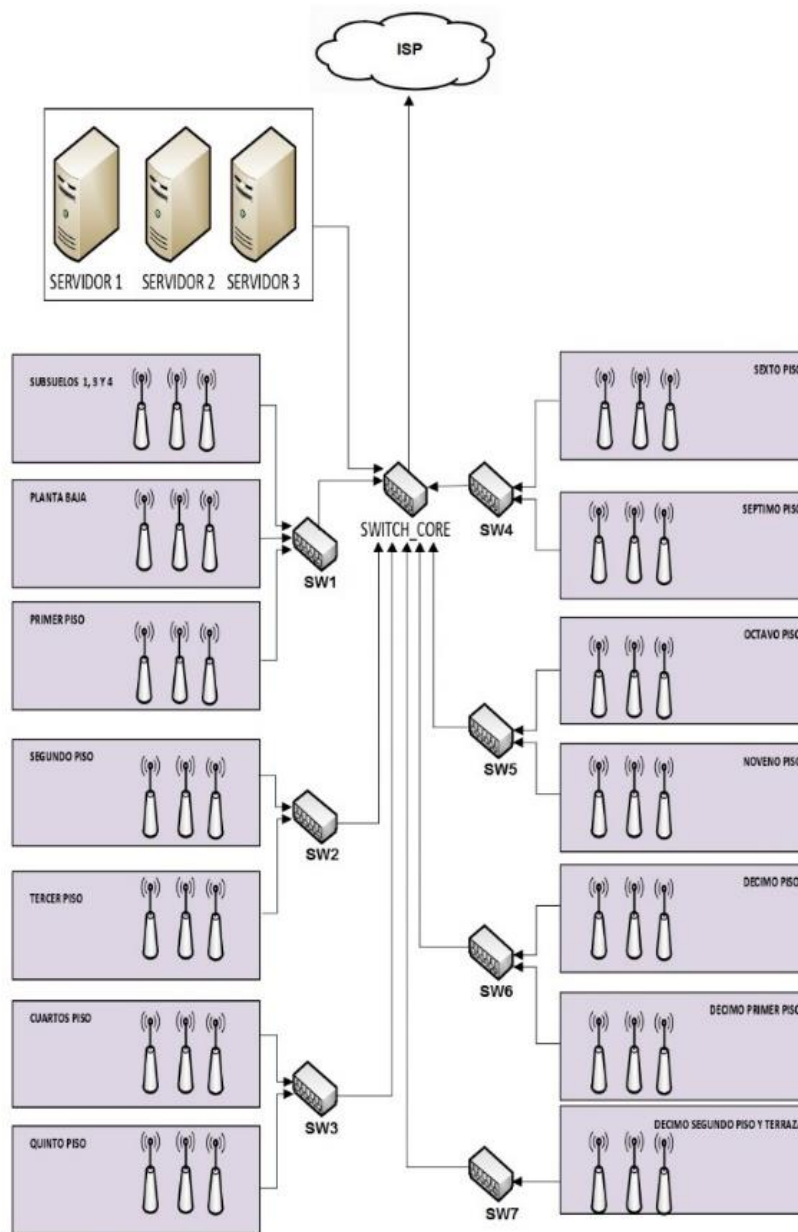
Debido a que la red del edificio necesitará únicamente una entidad gestora para evitar latencias y tener un diseño jerárquico que será manejada por el administrador se elige una arquitectura centralizada.

#### **3.4.2. Topología Lógica de la red WLAN**

Para el diseño se escogió la topología en estrella extendida debido a las ventajas que presenta como una latencia menor y su facilidad para implementar nuevos equipos en caso de ser necesario, tiene reducción menor de fallos en la red y se tomó en cuenta que los equipos que se encuentran en capa 2 o de enlace de datos se encargan de conectar los diferentes puntos de acceso con el servidor conectado al switch capa 3 y con ello su acceso a internet mediante un enlace con el ISP, en la figura 3.3 se visualiza la topología elegida.

Se descarta el uso de las topologías anillo y bus, debido a que dependen de una única vía de conexión para establecer una comunicación, lo cual las hace susceptibles a que, si falla un enlace, la red deja de funcionar completamente por lo cual las diferentes topologías analizadas para este diseño se presentan con mayor detalle en el anexo 7.

Figura 3.3 Topología lógica de la Edificación



Topología lógica de la edificación. Elaborado por: Chinachi Andrés, Rios Jorge.

### 3.4.3. Parámetros de configuración

#### 3.4.3.1. VLAN's

El uso de VLAN's permite cubrir las áreas de forma segmentada, a partir de los puntos de acceso o APs y el sitio en el que se encuentran, para ello se toma en cuenta un crecimiento del 25% de dispositivos sensores para los próximos cinco años para cada piso del edificio desde la planta baja, considerando los subsuelos como uno solo por los pocos nodos sensores que se instalarán en este y uniendo por secciones como se ve en la tabla 3.8.

Tabla 3.8 Planificación de crecimiento

PLANTA	Número de dispositivos	Total por sección	Crecimiento de 25% en dispositivos para 5 años	Total por sección en 5 años
Subsuelos	10	14	3	18
Terraza	4		1	
Planta Baja	29	29	7	36
Piso 1	36	594	9	748
Piso 2	42		11	
Piso 3	54		14	
Piso 4	54		14	
Piso 5	54		14	
Piso 6	54		14	
Piso 7	54		14	
Piso 8	54		14	
Piso 9	54		14	
Piso 10	54		14	
Piso 11	54		14	
Piso 12	30		8	
Elementos red WLAN	120	120	No aplica	No aplica
Servidores	3	3	No aplica	No aplica

Tabla de planificación de crecimiento de dispositivos en los próximos 5 años en el edificio Saint Charbel. Elaborado por: Chinachi Andrés, Rios Jorge.

Se eligió una red clase B que permite tener las características necesarias para redes medianas, y su direccionamiento se muestra en la tabla en la tabla 3.9.

Tabla 3.9 Direccionamiento de las vlan's del edificio Saint Charbel

VLAN	Nombre	Número de direcciones usadas	Número de direcciones máxima	Dirección	Sufijo	Dispositivos para cubrir
10	Departamentos	594	1022	172.16.0.0	22	Piso 1 al 12
20	Espacios	29	62	172.16.4.128	26	Planta baja
30	Equipos	14	30	172.16.4.192	27	Subsuelos y Terraza
40	Servidores	3	6	172.16.4.224	29	Servidores
100	Managment	120	126	172.16.4.0	25	Equipos de red
-	-	2	2	172.16.4.232	30	Enlace ISP

Tabla de direccionamiento de vlan's en el edificio Saint Charbel. Elaborado por: Chinachi Andrés, Rios Jorge.

### 3.4.3.2. Planeación de direcciones Lógicas

Se ha dividido los SSID como se muestra en la tabla 3.10, tres SSID para los diferentes sitios del edificio, tomando en cuenta que en cada planta habrá solo un SSID para reducir el retardo de paquetes, mientras que en la tabla 3.11 se observa las direcciones de los equipos de red.

Tabla 3.10 Distribución de SSID

Sitios	SSID	Número de puntos de acceso	Direcciones
Piso 1 al 12	Departamentos	99	172.16.4.11-172.16.4.99
Planta baja	Espacios	5	172.16.4.100-172.16.4.104
Subsuelos y terraza	Equipos	4	172.16.4.105-172.16.4.108

Tabla de distribución de SSID. Elaborado por: Chinachi Andrés, Rios Jorge.

Tabla 3.11 Direcciones de los equipos de red

Equipo	Etiqueta	Dirección	Puertos usados
Wireless Lan Controller	WLC	172.16.4.9/25	1
Servidor protocolo de mensajería	Servidor1	172.16.4.193 /29	1
Servidor plataforma IoT	Servidor2	172.16.4.194 /29	1
Servidor base de datos	Servidor3	172.16.4.195 /29	1

Tabla de direcciones de equipos de red. Elaborado por: Chinachi Andrés, Rios Jorge.

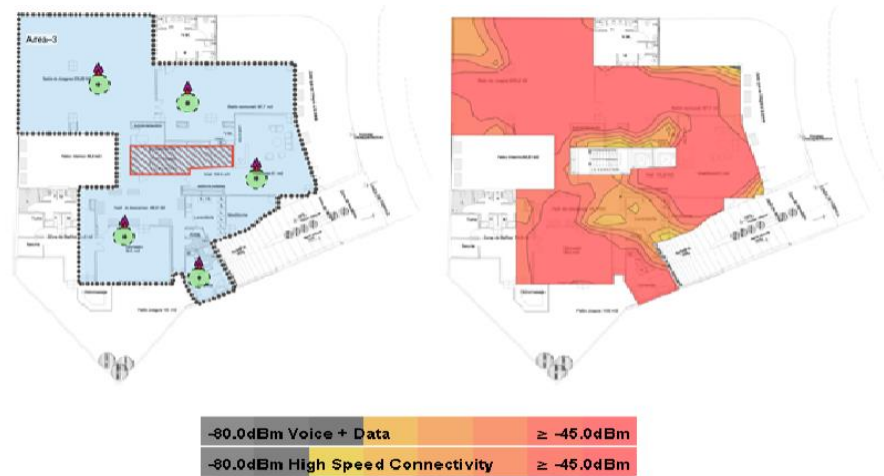
### 3.4.3.3. Mapas de Calor

Para establecer los mapas de calor se usó el software Ekahau el cual permite realizar un site survey pasivo basado en la distribución de los puntos de acceso en los diferentes espacios del edificio en donde estarán los sensores. Para ello se ingresó el mapa arquitectónico de los diferentes pisos y subsuelos de todo el edificio, se toma en cuenta el material del edificio y los posibles obstáculos para la propagación de la señal como son: paredes de bloques de concreto, ventanas gruesas, techo falso, muros de cemento, ascensores, armarios, puertas de madera gruesa, entre otros. Los parámetros a los que se realizan los mapas de calor son establecidos para tener una funcionalidad de alta velocidad teniendo como límite de intensidad de señal el valor de -70dbm, el cual es ideal para servicios como email o navegación.

Una vez establecidos los escenarios de cada piso, se elige las características de los APs, se realiza la distribución de canales, la potencia de transmisión, el número de Puntos de Acceso, ubicación adecuada y finalmente banda de frecuencia a la que se va a trabajar. Se realizan varias simulaciones para obtener los mapas de calor de los diferentes pisos, para la planta baja se colocan 5 puntos de acceso que permiten una cobertura óptima para las áreas en donde se prevé colocar los sensores como se ve en la figura 3.4, las simulaciones de los diferentes pisos se detallan en el anexo 8



Figura 3.4 Mapa de intensidad de señal de la Planta Baja



Ubicación de los puntos de acceso en la Planta Baja, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Ríos Jorge.

#### 3.4.4. Selección de equipamiento para la WLAN

Para realizar la selección del equipamiento necesario para la Red WLAN, se consideraron dispositivos para redes de pequeña y mediana escala usando como referencia al cuadrante de Gartner (ver anexo 9) para infraestructuras de acceso LAN cableado e inalámbrico, en el cual, se presentan las marcas con mayor posicionamiento en el mercado, desarrollo tecnológico y confiabilidad. Se evaluaron los fabricantes más importantes como Cisco, Huawei, Extreme Networks, Aruba y Juniper.

Para la elección de los APs de la red, se evalúa cuatro modelos de diferentes fabricantes para redes inalámbricas con sus respectivas características, mismas que se presentan en el anexo 10. El equipo seleccionado es de la marca Cisco modelo Aironet 702i debido a su bajo costo y características como banda dual de 2,4GHz y 5 GHz, un óptimo rendimiento y cobertura para interiores y tiene capacidad de alimentación mediante PoE. Los puntos de acceso son configurados mediante un controlador WLC (Wireless LAN Controller) Cisco modelo 3504 que permite la configuración del número de APs necesarios para la edificación y cuyos detalles y evaluación con respecto a otros modelos Cisco se presentan en el anexo 10.

#### 3.4.5. Cálculo del ancho de banda

Para asegurar un flujo de información adecuado y lograr una baja latencia en la red, se debe dimensionar un ancho de banda apropiado, por tal, se ha realizado la medición

del tamaño de paquetes en el envío de datos de cada uno de los sensores, con la ayuda del sniffer Wireshark, el cual, mediante la captura de tráfico de la red, permite obtener información detallada de paquetes usando como filtro las direcciones IP de los sensores, logrando obtener los datos mostrados en la tabla 3.12.

Tabla 3.12 Tamaño de paquetes

Sensor	Tamaño de Paquetes
Humo (Sensor MQ-2)	591 bytes
Nivel (Sensor HC-SR04)	173 bytes
Temperatura y humedad (Sensor DHT22)	649 bytes
Fuego (Sensor KY-026)	415 bytes
Gas (Sensor MQ-6)	576 bytes

Tabla de tamaño de paquetes según el sensor. Elaborado por: Chinachi Andrés, Rios Jorge.

A partir del tamaño de los paquetes de cada sensor, se realizó el cálculo de capacidad de ancho de banda requerido para los mismos de acuerdo al tiempo de envío de la información, considerando que el sensor de temperatura y de nivel, envían datos actualizados cada minuto, mientras que los sensores de humo, fuego y gas, envían datos actualizados cada 15 segundos debido a su prioridad en cuanto a situaciones de riesgo.

Para la capacidad de ancho de banda se utiliza la ecuación 3.1.

$$C = \frac{\text{Tamaño de paquete}}{1 \text{ publicación}} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ publicación}}{\text{Tiempo de publicación}} [\text{bits/s}] \quad \text{Ec. (3.1)}$$

Dando como resultado los siguientes datos:

- Sensor de Humo:  $C = \frac{591 \text{ bytes}}{1 \text{ publicación}} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ publicación}}{15 \text{ s}} = 315,2 [\text{bits/s}]$

En el edificio se considera 423 sensores de humo distribuidos en las instalaciones, por tal, la capacidad de ancho de banda para la cantidad total de sensores de humo es igual a 133 kbps.

- Sensor de Nivel:  $C = \frac{173 \text{ bytes}}{1 \text{ publicación}} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ publicación}}{60 \text{ s}} = 23,066 [\text{bits/s}]$

En el edificio se considera 1 sensor de nivel en la cisterna, por tal, la capacidad de ancho de banda es la misma.

- Sensor de Fuego:  $C = \frac{415 \text{ bytes}}{1 \text{ publicación}} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ publicación}}{15 \text{ s}} = 221,333 \text{ [bits/s]}$

En el edificio se considera 109 sensores de fuego distribuidos en las instalaciones, por tal, la capacidad de ancho de banda para la cantidad total de sensores de fuego es igual a 24 kbps.

- Sensor de Gas:  $C = \frac{576 \text{ bytes}}{1 \text{ publicación}} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ publicación}}{15 \text{ s}} = 307,2 \left[ \frac{\text{bits}}{\text{s}} \right]$

En el edificio se consideraron 99 sensores de gas, por ello, la capacidad de ancho de banda para estas lecturas es igual a 30,41 kbps.

- Sensor de Temperatura

$$C = \frac{649 \text{ bytes}}{1 \text{ publicación}} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ publicación}}{60 \text{ s}} = 86,533 \text{ [bits/s]}$$

En el edificio se consideraron 5 sensores de temperatura dando un total de ancho de banda igual a 0,43266 kbps. De acuerdo al análisis de tráfico realizado, el requerimiento de ancho de banda de cada uno de los sensores y el valor total de ancho de banda requerido en toda la red se recopila en la tabla 3.13.

Tabla 3.13 Capacidad de Ancho de Banda

Sensor	Capacidad de ancho de banda total
Humo (Sensor MQ-2)	133 kbps
Nivel (Sensor HC-SR04)	0,023 kbps
Temperatura y humedad (Sensor DHT22)	0,432 kbps
Fuego (Sensor KY-026)	24 kbps
Gas (Sensor MQ-6)	30,41 kbps
<b>Total</b>	<b>187,865 kbps</b>

Tabla de Capacidad de ancho de banda requerido en la red. Elaborado por: Chinachi Andrés, Rios Jorge.

Al utilizar dispositivos de recursos limitados en consumo de energía, velocidad y ancho de banda, cuya arquitectura es ideal para redes de sensores de gran escalabilidad, los requerimientos de ancho de banda de los mismos representan un mínimo absoluto de datos transmitidos, por tal el valor total correspondiente al requerimiento de

capacidad de ancho de banda en la red que es de 187,865 kbps para los 637 sensores que componen la edificación, lo cual no representan un valor considerable.

### **3.5. Diseño de la Red WSN**

#### **3.5.1 Elección de Protocolo de Comunicación Inalámbrica**

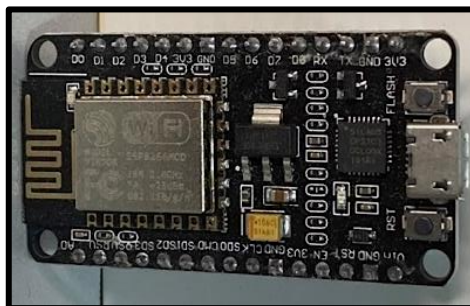
Se ha seleccionado el protocolo de comunicación inalámbrico Wi-Fi, debido a que permiten una tasa de transmisión mayor, es de fácil acceso, su rango de cobertura es ideal para interiores, además, según Gartner es la tecnología inalámbrica con mayor tiempo de vida y desarrollo en los últimos años, la tabla comparativa usada para la elección del protocolo de comunicación inalámbrica se presenta en el anexo 11. El estándar de Wi-Fi seleccionado corresponde al 802.11n usado para la transmisión y recepción de datos en la frecuencia de 2.4 GHz. Como parte de la infraestructura de red IoT, se considera la elección de una plataforma IoT que permita una gestión incorporando elementos como base de datos y protocolos de intercambio de mensajes que son instalados en servidores dedicados, formando parte de una solución.

#### **3.5.2. Equipamiento para la WSN**

##### **3.5.2.1. Elección de Motes o Nodos**

Para la elección de los nodos que serán los responsables de la comunicación y comando de los sensores hacia un gateway se consideran los controladores presentados en el anexo 12. El nodo seleccionado corresponde al NodeMCU versión 3 mostrado en la figura 3.5, debido a que es modular de tamaño reducido, accesible y económico e incorpora el protocolo de comunicación inalámbrico Wi-Fi, haciéndolo ideal para la comunicación de sensor a gateway sin la necesidad de un módulo especial que habilite la comunicación.

Figura 3.5 NodeMCU



Nodo controlador seleccionado. Elaborado por: Chinachi Andrés, Rios Jorge.

### 3.5.2.2. Fuentes de Alimentación

Los factores tomados en cuenta en la elección de la fuente de alimentación son los requerimientos para los cuales está diseñada la red, las plataformas que los nodos utilizan y el protocolo de comunicación inalámbrica se está usando. y debido a que esta red tiene que proporcionar servicios de comodidad y seguridad se establece que siempre debe estar activa conectándola a la red eléctrica del edificio y descartando fuentes como: baterías recargables, baterías solares, mediante la conexión USB, entre otros.

### 3.5.2.3. Elección de Sensores

Para la elección de los sensores que se utilizarán se han realizado tablas comparativas con las características más importantes de cada uno, presentadas en el anexo 12.

- **Sensores de Gas Butano y humo**

Se ha seleccionado un sensor para detectar gas GLP doméstico de modelo MQ-6 mostrado en la figura 3.6, debido a que su rango de detección de gas es adecuado para no confundirse con la presencia de otros gases en el aire. Para la detección de humo se considera al sensor de modelo MQ-2 ideal para detección de humo ante combustión, debido a su rango de operación el cual considera la concentración de humo en un sitio.

Figura 3.6 Sensor de humo y gas butano



Sensores de humo MQ-2 y gas butano MQ-6. Elaborado por: Chinachi Andrés, Rios Jorge.

- **Sensores de Humedad y Temperatura**

Se ha seleccionado el sensor DHT22 mostrado en la figura 3.7, debido a que es uno de los más completos ya que permite la detección tanto de humedad como de temperatura, además, es mayormente preciso y su rango de medición es completo.

Figura 3.7 Sensor de temperatura y humedad



Sensor DHT22 seleccionado. Elaborado por: Chinachi Andrés, Rios Jorge.

- **Sensor de Fuego**

Se selecciona el módulo KY-026 mostrado en la figura 3.8, sensor de llama único modular y de integración directa con diversos controladores de código abierto, su operación se basa en la detección de luz infrarroja emitida por el fuego, la cual tiene longitud de onda que puede ir de los 760 a los 1100 nm, siendo ideal para utilizarlo en sistemas contra incendios.

Figura 3.8 Sensor de fuego



SensorKY-026 seleccionado. Elaborado por: Chinachi Andrés, Rios Jorge.

- **Sensores de Nivel**

El sensor elegido es el HC-SR04 mostrado en la figura 3.9, debido a que es el sensor que tiene el mayor rango de distancia entre los sensores ultrasónicos, y es el único que tiene un ángulo de detección para mayor precisión.

Figura 3.9 Sensor de nivel



Sensor HC-SR04 seleccionado. Elaborado por: Chinachi Andrés, Rios Jorge.

### 3.6. Componentes IoT

#### 3.6.1. Elección de Plataforma IoT

En este proyecto se usó la plataforma OpenHAB, ya que existe una gran cantidad de soporte para las inquietudes y configuraciones, además, de ser compatible con la mayoría de los dispositivos en el mercado, facilita su automatización y permite mayor personalización e integración con diversos servidores externos, plataformas cloud y APIs. Su elección se basó en las principales características de las plataformas libres IoT como como DeviceHive, Thingsboard, Mainflux, Ubidots, Thingsboard, Thinger.io y Kaa que se puede ver en la tabla 3.14.

Tabla 3.14 Selección de Plataforma IoT de código abierto

Parámetros \ Plataformas IoT Open Source	DeviceHive	Thingsboard	Mainflux	Ubidots	OpenHAB	Assistant Home	Thinger.io	Kaa
Es gratis	x		x		x	x		
Servicios Cloud propio				x	x		x	x
Interfaces propias para dispositivos móviles				x	x	x		
Fácil adquisición de información para ayuda y soporte.	x	x	x	x	x		x	x
Alta interoperabilidad para gestión de dispositivos			x	x	x	x	x	x
Trabaja con bases de datos SQL, NoSQL e Híbridas	x	x	x	x	x	x	x	x
Integración con plataformas y servicios en la nube	x	x	x	x	x	x	x	x
Interfaces personalizables para la visualización y recopilación de datos	x	x	x	x	x	x	x	x
Trabaja con protocolos IoT (multiprotocolo) y APIs	x	x	x	x	x	x	x	x
Permite la automatización y creación de reglas, alarmas	x	x	x	x	x	x	x	x
Paneles en tiempo real	x	x	x	x	x	x	x	x
Tolerancia a fallos	x	x	x	x	x	x	x	x
Varias opciones de despliegue	x	x	x	x	x	x	x	x
<b>Selección</b>								

Tabla de selección y comparación de plataforma IoT. Elaborado por: Chinachi Andrés y Rios Jorge.

El primer punto para su selección es que la plataforma OpenHAB es totalmente gratuita, permite integrar varios dispositivos y ofrecer todos los servicios que posee sin ningún costo frente a plataformas como Ubidots, Thingsboard, Thinger.io, que son muy populares y confiables en el mercado, sin embargo, el costo por sus servicios depende del número de dispositivos a integrar o de los servicios que necesitará. Como punto fuerte diferenciador se tiene que para el acceso remoto de los servicios que prestan, las únicas plataformas analizadas que tienen un servicio de Cloud propio son Thinger.io, Ubidots, OpenHAB y Kaa destacando OpenHAB de igual manera debido a que este servicio es gratuito. Las únicas plataformas que poseen interfaces móviles

propias para los teléfonos inteligentes de las analizadas, son: Ubidots, OpenHAB y Home assistant, las demás necesitan la integración con otras plataformas y APIs para el envío de notificaciones y alertas. De las seis plataformas analizadas únicamente Home Assistant tiene poca cantidad de información para ayuda y soporte con respecto a las demás, esto es principalmente debido a que es una plataforma que lleva pocos años y la información es todavía muy limitada en cuanto a configuraciones y soporte. Todas las plataformas tienen gran compatibilidad con los dispositivos en el mercado, sin embargo, según información de sus sitios Web son Mainflux, Ubidots, OpenHAB, Home Assistant, Thingier.io y Kaa son las que poseen mayor interoperabilidad con diferentes dispositivos.

Las seis plataformas analizadas admiten trabajar con multiprotocolos IoT, análisis en tiempo real de los datos mediante gráficas, trabajar con APIs para la visualización de datos y para el envío de alertas o notificaciones a otras plataformas, admiten varias bases de datos disponibles tanto SQL, no SQL o Híbridas haciéndolas amigables para la forma en que se desee guardar los datos, varias opciones de despliegue ya que pueden implementarse a varias escalas desde una Raspberry Pi hasta servidores de datos según las necesidades, de igual manera todas son personalizables ya que ocupan herramientas propias para la visualización y recopilación de datos de una forma visualmente atractiva, lo que les hace a todas aptas para trabajar con redes IoT. Las plataformas que más pueden competir con OpenHAB según el presente análisis son Ubidots y Home Assistant ya que tienen cualidades muy similares y puntos fuertes a su favor como los mencionados anteriormente.

### **3.6.2. Elección de Protocolo de intercambio de mensajes**

El protocolo seleccionado para el intercambio de mensajes es MQTT, que permite trabajar sobre TCP/IP, tener mejor compatibilidad con dispositivos existentes y escalabilidad, su selección y comparación, se basa en las principales características de los diferentes protocolos de intercambio de mensajes y en su compatibilidad con plataformas IoT, que se puede observar en el anexo 13.



### 3.6.3. Elección de Base de Datos

Se seleccionó la base de datos InfluxDB, debido a que integra una mayor compatibilidad, permite un almacenamiento de datos basado en lenguaje SQL y es ideal para proyectos de sensores IoT por su rendimiento y consultas en tiempo real, además que incorpora un servicio de gestión de gráficas y monitoreo mediante la plataforma Grafana, su selección y comparación se puede ver en el anexo 14.

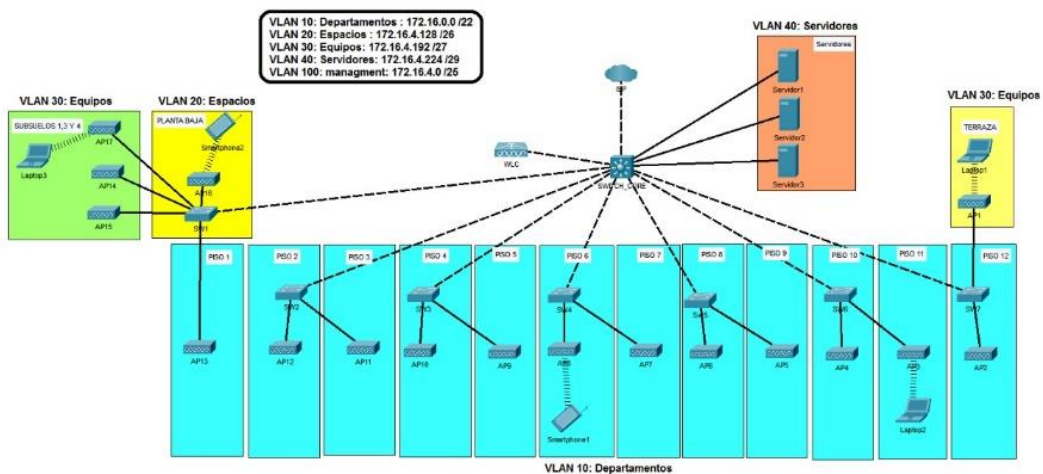
### 3.6.4. Elección de Servidores

Para la gestión de cada uno de los sensores que componen la red WSN, es necesario la elección de una plataforma IoT complementada con una base de datos y un protocolo de intercambio de mensajes, los cuales son instalados en servidores dedicados para brindar soporte a la demanda de tráfico en la red, los servidores seleccionados corresponden a la marca DELL modelo PowerEdge R240, el cual posee características más que suficientes y con un amplio almacenamiento a un costo accesible y con gran poder de cómputo para la red. La elección de servidores se muestra en el anexo 15.

### 3.7. Simulaciones

Para la red WLAN se usa el software Packet Tracer debido a que permite tener una interfaz similar a la real en la controladora WLC para los puntos de acceso. Para la verificación de su funcionamiento y conectividad se realiza una simulación presentada en la figura 3.10, las configuraciones de la controladora se pueden observar en el anexo 16.

Figura 3.10 Diagrama de la simulación para la WLAN



Simulación para la red WLAN en la edificación. Elaborado por: Chinachi Andrés y Rios Jorge

### 3.7.1. Configuración de los dispositivos

Para la configuración y simulación se usó el software Packet Tracer. Los switches de cada piso tienen una configuración similar cambiando únicamente su dirección de administración, la configuración de los APs se realizó mediante una controladora.

- **Configuración APs**

Para configurar los puntos de acceso se utiliza la controladora 3504 de Cisco, la cual primero se registra con una dirección estática para su administración vinculada a la vlan 100, se configura las tres interfaces para las VLANs 10, 20, 30 con sus respectivas direcciones para enlazarla con el switch de core en el que se crearon los pools de direcciones para el DHCP.

- **Configuración WLC**

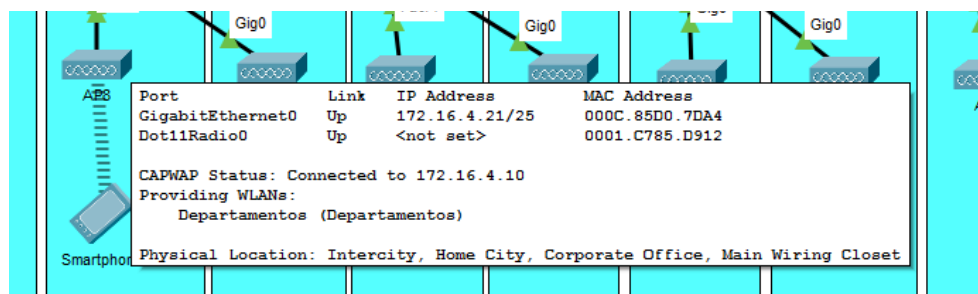
Se configura el controlador de LAN inalámbrica con tres WLAN establecidas con los siguientes nombres: Departamentos, Equipos y Espacios, las cuales utilizan seguridad WPA2, se configura interfaces de administración, se establece los SSID y la contraseña, y en las opciones se establece la comunicación FlexConnect y la autenticación local FlexConnect. Finalmente se crea un grupo de APs según la WLAN, en donde se vinculan los puntos de acceso registrados en la controladora, los cuales tienen una dirección de administración de la vlan 100.

### 3.7.2. Resultados

- **Conectividad en la red**

Los APs tienen direcciones de administración de la vlan 100 como se ve en la figura 3.11, y se vinculan a la WLAN creada a la controladora.

Figura 3.11 Dirección del AP8 correspondiente al piso 6



Información del AP8 en el piso 6. Elaborado por: Chinachi Andrés y Rios Jorge.

Se comprueba la asignación dinámica DHCP para los clientes, mediante la colocación de teléfonos, smartphones y laptops, vinculándolos a los SSIDS creados, además, se utiliza el comando ping para demostrar la conectividad con el Gateway que es el switch de core, dichas comprobaciones se pueden observar en las figuras 3.12, 3.13 y 3.14.

Figura 3.12 Dirección obtenida mediante DHCP para la laptop 2

```
C:\>ipconfig

Wireless0 Connection: (default port)

    Link-local IPv6 Address . . . . . : FE80::2E0:B0FF:FEAC:7479
    IP Address . . . . . : 172.16.0.2
    Subnet Mask . . . . . : 255.255.252.0
    Default Gateway . . . . . : 172.16.0.1

C:\>ping 172.16.0.1

Pinging 172.16.0.1 with 32 bytes of data:

Reply from 172.16.0.1: bytes=32 time=96ms TTL=255
Reply from 172.16.0.1: bytes=32 time=6ms TTL=255
Reply from 172.16.0.1: bytes=32 time=7ms TTL=255
Reply from 172.16.0.1: bytes=32 time=7ms TTL=255

Ping statistics for 172.16.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 6ms, Maximum = 96ms, Average = 29ms
```

Dirección obtenida mediante DHCP para la laptop 2 conectado al SSID “Departamentos” y conexión exitosa hacia el switch de core. Elaborado por: Chinachi Andrés y Rios Jorge.

Figura 3.13 Dirección obtenida mediante DHCP para la laptop 1

```
C:\>ipconfig

Wireless0 Connection: (default port)

    Link-local IPv6 Address . . . . . : FE80::201:42FF:FEA0:C31A
    IP Address . . . . . : 172.16.4.194
    Subnet Mask . . . . . : 255.255.255.224
    Default Gateway . . . . . : 172.16.4.193

C:\>ping 172.16.4.193

Pinging 172.16.4.193 with 32 bytes of data:

Reply from 172.16.4.193: bytes=32 time=32ms TTL=255
Reply from 172.16.4.193: bytes=32 time=10ms TTL=255
Reply from 172.16.4.193: bytes=32 time=115ms TTL=255
Reply from 172.16.4.193: bytes=32 time=76ms TTL=255

Ping statistics for 172.16.4.193:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 10ms, Maximum = 115ms, Average = 58ms
```

Dirección obtenida mediante DHCP para la laptop 1 conectado al SSID “Equipos” y conexión exitosa hacia el switch de core. Elaborado por: Chinachi Andrés y Rios Jorge.

Figura 3.14 Dirección obtenida mediante DHCP para el Smartphone 2

```
C:\>ipconfig

Wireless0 Connection:(default port)

Link-local IPv6 Address.....: FE80::230:A3FF:FEE2:9CB
IP Address.....: 172.16.4.131
Subnet Mask.....: 255.255.255.192
Default Gateway.....: 172.16.4.129

C:\>ping 172.16.4.129

Pinging 172.16.4.129 with 32 bytes of data:

Reply from 172.16.4.129: bytes=32 time=201ms TTL=255
Reply from 172.16.4.129: bytes=32 time=8ms TTL=255
Reply from 172.16.4.129: bytes=32 time=38ms TTL=255
Reply from 172.16.4.129: bytes=32 time=7ms TTL=255

Ping statistics for 172.16.4.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 7ms, Maximum = 201ms, Average = 63ms
```

Dirección obtenida mediante DHCP para el Smartphone 2 conectado al SSID “Espacios” y conexión exitosa hacia el switch de core. Elaborado por: Chinachi Andrés y Rios Jorge

### 3.8. Análisis Económico

#### 3.8.1. Costos de diseño de hardware

Los costos de diseño de hardware comprenden, el diseño de conexión de los nodos con sus respectivos sensores y actuadores, y el costo de mano de obra por cada hora de servicio prestado, los cuales se muestran en la tabla 3.15.

Tabla 3.15 Costo diseño de hardware

Descripción	Cantidad (personas)	Tiempo (horas)	Precio (Hora)	Precio total del trabajo (\$)
Diseño de Ingeniería	2	15	5,00	\$150,00
Mano de obra	2	144	5,00	\$1.440,00
<b>Total</b>				<b>\$1.590,00</b>

Tabla de costos por hora de diseño de hardware. Elaborado por: Chinachi Andrés y Rios Jorge.

#### 3.8.2. Costos de diseño de software

El costo de diseño de software se detala en la tabla 3.16, considerando cada hora de trabajo necesario para el desarrollo del software.

Tabla 3.16 Costo diseño de software

Descripción	Cantidad (personas)	Tiempo (horas)	Precio (Hora)	Precio total del trabajo (\$)
Programación NodeMCU	2	3	5,00	\$30,00
Programación plataforma IoT	2	7	5,00	\$70,00
Programación base de datos	2	1	5,00	\$10,00
Configuración equipos WLAN	2	100	5,00	\$1.000,00
Diseño de Ingeniería	2	80	5,00	\$800,00
<b>Total</b>				<b>\$1.910,00</b>

Tabla de costos por hora de diseño de software. Elaborado por: Chinachi Andrés y Rios Jorge.

### 3.8.3. Costos de implementación

Al realizar un análisis y elección de cada uno de los elementos que componen la red, se plantea un presupuesto detallado en la tabla 3.17, en el cual se detallan todos los costos que tendrá llevar a cabo su ejecución considerando equipos de WLAN, WSN, IoT y materiales necesarios para la comunicación de ambos, la mano de obra comprende la instalación de la red IoT, referenciando al trabajo de implementación por parte de los ingenieros, los costos de implementación llegan a tener un valor total de \$40.986,26.

Tabla 3.17 Resumen del costo de implementación

Nombre	Unidad	Cantidad	Precio Unitario	Precio Final
<b>Equipo WLAN</b>				
Access Point CISCO Aironet 702i	C/U	108	\$99,99	\$10.798,92
WLC CISCO 3504	C/U	1	\$1.590	\$1.590,00
<b>Equipo WSN</b>				
Node MCU V3	C/U	637	\$7	\$4.459,00
Sensor MQ-2	C/U	423	\$5	\$2.115,00
Sensor MQ-6	C/U	99	\$5	\$495,00
Sensor DHT22	C/U	5	\$10	\$50,00
Sensor KY-026	C/U	109	\$2,70	\$294,30
Sensor HC-SR04	C/U	1	\$3,60	\$3,60
Buzzer	C/U	632	\$2	\$1.264,00
<b>Equipo IoT</b>				
Servidor DELL PowerEdge R240	C/U	3	\$800	\$2.400,00
<b>Materiales de cableado estructurado/LAN</b>				
Rack de pared 19" de 6 unidades	C/U	6	\$125	\$750,00
Patch Panel	C/U	8	\$75	\$600,00
Bandeja para Rack	C/U	11	\$15	\$165,00
Organizadores	C/U	8	\$17	\$136,00
Tapa Ciega	C/U	24	\$7	\$168,00
PDU Monofásico Horizontal	C/U	7	\$75	\$525,00
Escalerilla 8"x10'	M	12	\$27	\$324,00
Cable UTP Cat.6A	Rollo	8	\$280	\$2.240,00
Conectores Rj45 Cat. 6	Funda	7	\$6,50	\$45,50
Manguera Bx 1"1/4'	M	17	\$130	\$2.210,00
Canaleta plástica de pared	M	6	\$1,50	\$9,00
Crimpeadora	C/U	1	\$14	\$14,00
Elementos Varios		1	\$1.000	\$1.000,00
<b>Equipo LAN</b>				
Switch CISCO capa 3 mb Sg250-18	C/U	1	\$419,99	\$419,99
Switch CISCO capa 2 Sg220-28mp-k9-na	C/U	7	\$772,85	\$5.409,95
<b>Mano de obra</b>				
Costos Diseño de Hardware y Software				\$3.500
<b>Total</b>				\$40.986,26

Tabla de resumen de costos de implementación. Elaborado por: Chinachi Andrés y Rios Jorge.

### 3.8.4. Gastos de operación

Los gastos de operación en este proyecto, comprenden el costo para poner en marcha la operación de la Red IoT, considerando que el servicio requiere de un plan de Internet para su funcionamiento, independiente de la red GPON; se considera un gasto inicial correspondiente a la elaboración del presente estudio para puesta en marcha de la red; arriendo de servicio básico de luz necesaria para alimentar los dispositivos y finalmente el costo de mantenimiento, el cual es mensual y se paga por horas trabajadas a los técnicos. En la tabla 3.18 se detalla la proyección de gastos de operación para los próximos tres años.

Tabla 3.18 Gastos de operación

<b>Gastos de operación</b>					
	<b>Mes</b>	<b>Año 0</b>	<b>Año 1</b>	<b>Año 2</b>	<b>Año 3</b>
Servicio de luz eléctrica	\$1.500,00	\$18.000,00	\$18.000,00	\$18.000,00	\$18.000,00
Servicio de internet	\$27,89	\$334,68	\$334,68	\$334,68	\$334,68
Elaboración del estudio		\$450,00	-	-	-
Costo de implementación		\$3.500,00	-	-	-
Mantenimiento	\$50,00	\$600,00	\$600,00	\$600,00	\$600,00
<b>Gastos Totales</b>		<b>\$22.884,68</b>	<b>\$18.934,68</b>	<b>\$18.934,68</b>	<b>\$18.934,68</b>

Tabla de gastos de operación en un periodo de 3 años. Elaborado por: Chinachi Andrés y Rios Jorge.

### 3.8.5. Valor actual neto (VAN)

El cálculo del VAN, permite conocer valor de ganancia o pérdida de una inversión y el efecto que puede causar sobre una empresa, para el cálculo se utiliza la ecuación 3.2, donde:  $F_t$  es el flujo de caja,  $I_0$  inversión inicial, n número de periodos de tiempo y k la tasa de descuento.

$$VAN = -I_0 + \sum_{t=1}^n \frac{F_t}{(1+k)^t} \quad \text{Ec. (3.2)}$$

Se considera una tasa de descuento referencial publicado por el Banco Central del Ecuador con fecha de corte en agosto del 2020 con un valor de 10,19%, los valores considerados para el cálculo se indican en la tabla 3.19.

Tabla 3.19 Cálculo de VAN

Variable	Valor
$I_0$	\$40.986,26
$F_t$	\$24.000,00
n	2 años
k	10,19%
<b>VAN</b>	\$560,67

Tabla de valores utilizados para el cálculo del VAN. Elaborado por: Chinachi Andrés y Rios Jorge.

Al ser el VAN mayor que cero, se indica que el proyecto es viable para la Constructora SAUD.

### 3.8.6. Tasa interna de retorno (TIR)

La tasa de rentabilidad que ofrece la inversión se calcula mediante la ecuación 3.3, donde  $F_t$  es el flujo de caja,  $I_0$  inversión inicial y n el número de periodos de tiempo, los valores considerados para el cálculo se presentan en la tabla 3.20.

$$0 = -I_0 + \sum_{t=1}^n \frac{F_t}{(1+TIR)^t} \quad \text{Ec. (3.3)}$$

Tabla 3.20 Cálculo de TIR

Variable	Valor
$I_0$	\$40.986,26
$F_t$	\$24.000,00
n	2 años
<b>TIR</b>	11%

Tabla de valores utilizados para el cálculo del TIR. Elaborado por: Chinachi Andrés y Rios Jorge.

El TIR es mayor que la tasa de retorno, por tal el proyecto es totalmente rentable y para la constructora representa un valor que obtendrá ganancias conjuntamente con la venta, alquiler de locales comerciales y administración de los departamentos de la edificación.

### 3.8.7. Periodo de recuperación de inversión (PRI)

El PRC indica el periodo en el cual se da una recuperación del valor de inversión del proyecto, se calcula mediante la ecuación 3.4, donde a es el último año de flujo negativo, b es la inversión inicial, c es el flujo de efectivo acumulado del último año

de flujo negativo, d es el primer flujo de caja positivo. Los valores obtenidos se presentan en la tabla 3.21.

$$PRI = a + \frac{(b-c)}{d} \quad \text{Ec. (3.4)}$$

Tabla 3.21 Cálculo de PRI

Variable	Valor
<i>a</i>	1
<i>b</i>	\$40.986,26
<i>c</i>	\$16.986,26
<i>d</i>	\$24.000,00
<b>PRI</b>	2 años

Tabla de valores utilizados para el cálculo del PRI. Elaborado por: Chinachi Andrés y Rios Jorge.

El PRI indica que el valor de la inversión se recupera en un periodo de dos años.

### 3.8.8. Relación beneficio/costo

La relación beneficio costo, permite determinar si un proyecto es totalmente viable, se calcula mediante la ecuación 3.5 y los valores obtenidos Durante los próximos tres años se presenta en la tabla 3.22.

$$B/C = \frac{\text{Ingresos actualizados}}{\text{Costos actualizados}} \quad \text{Ec. (3.5)}$$

Tabla 3.22 Relación beneficio/costo

Relación beneficio/costo		
Años	Ingresos actualizados	Costos actualizados
0	\$23.520,00	\$22.884,68
1	\$23.520,00	\$18.934,68
2	\$23.520,00	\$18.934,68
3	\$23.520,00	\$18.934,68
<b>Total</b>	\$94.080,00	\$79.688,72
<b>B/C</b>	1,18059369	

Tabla relación beneficio/costo en un periodo de 3 años. Elaborado por: Chinachi Andrés y Rios Jorge.



### 3.8.9. Análisis

En la tabla 3.23, se presenta un resumen de los cálculos obtenidos para determinar la viabilidad del proyecto, a partir de lo cual se puede determinar que:

El van al ser mayor que cero, indica que el proyecto es conveniente, caso contrario no sería viable, para el caso del proyecto se obtiene un valor de \$560,67. El TIR es de 11% lo que indica que los flujos generados por el proyecto son capaces de recuperar la inversión de \$40.986, 26, en un periodo de 2 años; finalmente la relación costo/beneficio, al ser mayor que la unidad, indica que la recuperación de la inversión efectivamente ocurre y que es factible ejecutar el proyecto.

Tabla 3.23 Tabla resumen

VAN	\$560,67
TIR	11%
PRI	2 años
B/C	1,18059369

Tabla de resumen de escenario obtenido en el proyecto. Elaborado por: Chinachi Andrés y Ríos Jorge.

## CAPÍTULO 4

### 4. SIMULACIÓN Y PRUEBAS

#### 4.1. Generalidades

Con el fin de probar la red WSN y su integración con la plataforma IoT Open Source, se realizó la construcción de una maqueta que permite diseñar un ambiente práctico para demostrar el funcionamiento de los nodos sensores y los elementos que se involucran en su vinculación a la plataforma IoT Open Source y su administración en la nube.

Para tener una muestra que permita generalizar a toda la población de sensores especificados y sea una representación a escala de esta, se considera la ecuación 4.1 (Urbina, 2016) para una muestra finita, tomando en cuenta los siguientes parámetros: Población  $N$  de 637 sensores, la estimación de la proporción que se desea medir  $p$  el cual se considera del 99% ya que se quiere probar la mayoría de los tipos de sensores escogidos, un margen de confianza del 95% correspondiente a un  $Z=1.96$ , con un margen de error  $e$  de 9% que es considerado bueno.

$$n = \frac{N*Z^2*p*q}{e^2*(N-1)+Z^2*p*q} \quad \text{Ec. (4.1)}$$

$$N = 637, p = 99\% = 0.99, Z = 1.96, e = 0.09$$

$$q = 1 - p = 1 - 0.99 = 0.01$$

$$n = \frac{637*1.96^2*0.99*0.01}{0.09^2*(637-1)+1.96^2*0.99*0.01} = 4.66 \approx 5$$

La muestra de sensores necesarias es de 5, por ello se toma en cuenta para la demostración un sensor de cada tipo de los establecidos anteriormente, con el fin de probar el funcionamiento e integración de la plataforma, para ello se realiza una representación de la planta baja del edificio, la cual tiene un dimensionamiento ideal para los espacios elegidos y adicional un tanque que permite simular la cisterna del mismo.

## 4.2. Configuración

### 4.2.1. Configuración de red WSN

Los sensores que conforman la red WSN, son comandados mediante el módulo controlador NodeMCU, el cual se ha programado en una plataforma Arduino IDE, para que dicha plataforma reconozca al módulo se agregó la fuente gestora de tarjetas proveniente de la página de dragino, e instaló la placa ESP8266. Adicional se instalaron las librerías para el funcionamiento de los sensores como DHT, Newping, PubSubClient y ESP8266WiFi, la programación del sensor de nivel se presenta en la figura 4.1, y el resto de sensores se detalla en el anexo 17.

Figura 4.1 Configuración sensor de nivel

```
1. #include <NewPing.h>
2. #include <ESP8266WiFi.h>
3. #include <PubSubClient.h>
4. #define wifi_ssid "XXXXXX"
5. #define wifi_password "XXXXXX"
6. #define mqtt_server "192.168.1.17"
7. #define mqtt_user "administrador"
8. #define mqtt_password "XXXXXX"
9. #define
   distancia_topic "Edificio/Planta Baja/Subsuelo2/sensor/ul53.
   trasonico"
10. #define TRIGGER_PIN 0
11. #define ECHO_PIN 5
12. #define MAX_DISTANCE 900
13. #define myClientID "nivel 1"
14. #define myUsername "Sensor_nivel_subsuelo2"
15. #define myPassword "12345"
16. double porcenminimo = 90;
17. char message_buff[100];
18. long lastMsg = 0;
19. long lastRecu = 0;
20. bool debug = false;
21. int buzzer = D7;
22. int LED_PIN = D5;
23. WiFiClient espClient;
24. PubSubClient client(espClient);
25. NewPing sonar(TRIGGER_PIN, ECHO_PIN,
   MAX_DISTANCE);
26. void setup() {
27.   Serial.begin(9600);
28.   setup_wifi();
29.   pinMode(buzzer, OUTPUT);
30.   pinMode(LED_PIN, OUTPUT);
31.   client.setServer(mqtt_server, 1883);
32.   client.connect("myClientID", "myUsername", "myPassw
   ord");
33.   Serial.println("Sensor de nivel...");
34. }
35. void setup_wifi(){
36.   delay(10);
37.   Serial.println();
38.   Serial.print("Connecting to");
39.   Serial.println(wifi_ssid);
40.   WiFi.begin(wifi_ssid, wifi_password);
41.   while(WiFi.status() != WL_CONNECTED){
42.     delay(500);
43.     Serial.print(".");
44.   }
45.   Serial.println("");
46.   Serial.println("WiFi OK");
47.   Serial.println("=> ESP8266 IP address: ");
48.   Serial.print(WiFi.localIP());
49. }
50. void reconnect() {
51.   while(!client.connected()){
52.     Serial.print("Connecting to MQTT broker ...");
53.     if(client.connect("ESP8266Client", mqtt_user,
   mqtt_password)){
54.       Serial.println("OK");
55.     }else{
56.       Serial.print("OK, error. ");
57.       Serial.print(client.state());
58.       Serial.println("Wait 5 seconds before to retry");
59.       delay(2000);
60.     }
61.   }
62. }
63. void loop() {
64.   if(!client.connected()){
65.     reconnect();
66.   }
67.   client.loop();
68.   double d=sonar.ping_cm();
69.   double por=map(d,0,200,100,0);
70.   if (d > porcenminimo) {
71.     digitalWrite(buzzer, HIGH);
72.     digitalWrite(LED_PIN, HIGH);
73.     delay(500);
74.   } else {
75.     digitalWrite(buzzer, LOW);
76.     digitalWrite(LED_PIN, LOW);
77.     delay(200);
78.   }
79.   long now=millis();
80.   if(now - lastMsg > 60000){
81.     lastMsg = now;
82.     Serial.print("PORCENTAJE: ");
83.     Serial.print(String(por).c_str());
84.     Serial.println(" %");
85.     Serial.print("valor en cm");
86.     Serial.println(String(d).c_str());
87.     client.publish(distancia_topic, String(por).c_str(), true);
88.   }
89. }
```

Programación sensor de nivel. Elaborado por: Chinachi Andrés y Rios Jorge.

### 4.2.2. Simulación y configuración de servidores

#### 4.2.2.1. Protocolo para el intercambio de mensajes

Para la implementación del protocolo de comunicación MQTT, se realizó la instalación del bróker servidor MQTT Mosquitto, utilizando el software VirtualBox que permite virtualizar el mismo y de esta forma acceder a los servicios que MQTT ofrece a la red, se utiliza la distribución Ubuntu 20.04 LTS de Linux como sistema

operativo. La instalación del servidor se realizó por líneas de comando en el terminal de Ubuntu, comenzando por instalar los repositorios de Mosquitto, necesarios para permitir la descarga e instalación del mismo en la máquina, se instaló el bróker, la librería y el cliente MQTT, para dar seguridad a la comunicación del servidor, se creó un usuario con contraseña en Mosquitto y finalmente se puso en funcionamiento el servidor, se configuró el usuario y contraseña MQTT en cada uno de los sensores a utilizar para establecer una comunicación con el mismo.

#### **4.2.2.2. Base de Datos**

La instalación de la base de datos se la realizó en la distribución Ubuntu 20,04 LTS de Linux mediante el software VirtualBox, agregando los repositorios necesarios para instalar la base de datos y habilitando los servicios de autenticación que permitan acceder al puerto 8086 para una comunicación del mismo con la plataforma.

La base de datos almacenará las lecturas de los sensores, para la creación de gráficas se usó la plataforma Grafana la cual se instala agregando los repositorios necesarios, se configura una iniciación automática para cada uno de los servidores, para de esta forma facilitar el arranque de los servicios al encender la máquina sin la necesidad de hacerlo manualmente, una vez instalado el servicio se accede a su interfaz de configuración web mediante el puerto 3000.

#### **4.2.2.3. Plataforma IoT**

OpenHAB es una plataforma de código abierto, que permite el monitoreo y control de dispositivos de más de 200 fabricantes. Se instaló una máquina virtual mediante el software VirtualBox sobre la distribución Ubuntu 20,04 LTS, usando Zulu que es una plataforma Java 8 Open Source, el cual es un prerequisite para el óptimo funcionamiento de openHAB. La instalación de OpenHAB versión 2.5.8, se realizó mediante el repositorio de paquetes utilizando el terminal de Ubuntu, una vez instalados todos los elementos se accede al navegador mediante la dirección `http://ip-de-virtual-box: 8080` y se seleccionó el paquete estándar que se observa en la figura 4.2.

Figura 4.2 Interfaz web openHAB

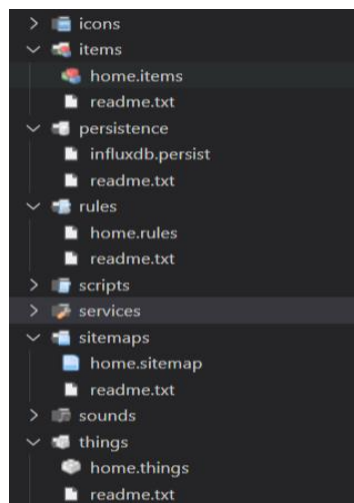


Paquete estándar de openHAB 2. Elaborado por: Chinachi Andrés y Rios Jorge.

Debido a que openHAB trabaja con diferentes archivos de configuración como son los ítems, things, sitemaps y rules, los cuales son archivos de configuración principales que permiten la automatización de los diferentes dispositivos, es necesario crear estos archivos en las carpetas del mismo nombre, para eso desde el terminal de Ubuntu se fue creando documentos de texto vacíos correspondientes a cada uno de los archivos antes mencionados.

Para tener la facilidad de configurar los archivos y carpetas que se encuentran en openHAB se instaló el software Samba que permite tener un recurso compartido de red para los archivos de openHAB, mientras que para acceder y configurar estos, se utilizó Visual Studio, debido a que openHAB cuenta con una extensión llamada openHAB VS que permite editar los archivos de configuración y se pueden observar en la figura 4.3.

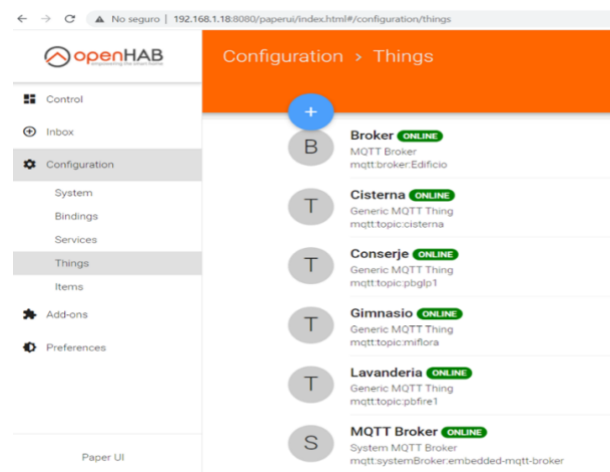
Figura 4.3 Archivos de configuración



Archivos de configuración creados. Elaborado por: Chinachi Andrés y Rios Jorge.

En visual studio se accede a los archivos de configuración de openhab, donde se creó las things que son la representación de los dispositivos sensores también llamados nodos o de un servicio vinculándolo a la plataforma, los cuales se crearon uno para cada nodo sensor adicionando las características y funcionalidades, además en este archivo se establece la conexión al servidor mqtt mediante el uso de un binding que permite enlazar los dispositivos físicos, con el sistema y elementos web que posee openHAB en su plataforma, como se ve en la figura 4.4, para ello es necesaria una integración MQTT cuya configuración se indica en la figura 4.5.

Figura 4.4 Things en openHAB



Things configuradas en openHAB. Elaborado por: Chinachi Andrés y Rios Jorge.

Figura 4.5 Archivo de configuración para Things

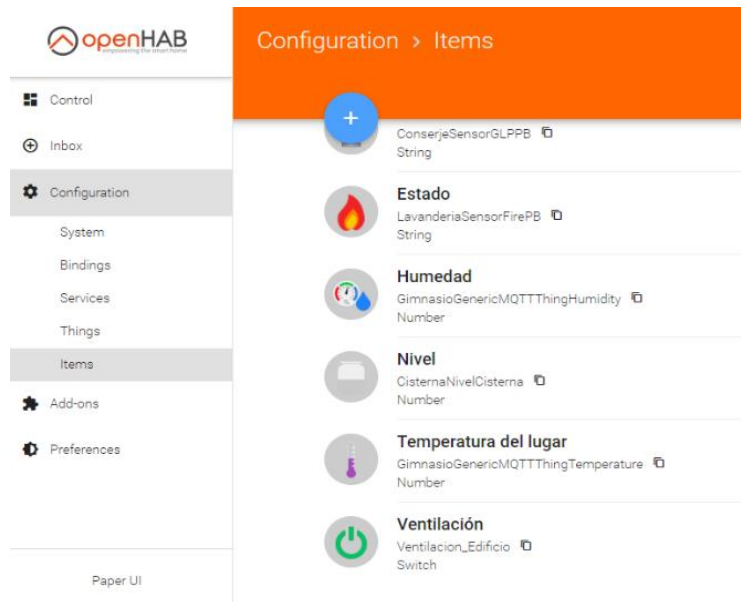
```
Bridge mqtt:broker:Edificio "Broker" @ "Planta Baja" [ host="192.168.1.17", secure=false, username="andres", passwd="admin" ]
{
  Thing mqtt:topic:cisterna "Cisterna" @ "Subsuelo2" {
    Channels:
      Type number : NivelCisterna "Nivel " [stateTopic="Edificio/Planta_Baja/Subsuelo2/sensor/ultrasonico"]
  }
}
```

Ejemplo de configuración del archivo things en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

- Integración MQTT: La integración con el servidor MQTT se realizó accediendo a la interfaz de la plataforma e instalando MQTT binding y MQTT broker Mosquitto en addons de la interfaz, además se configuró el MQTT broker con la dirección IP, usuario y contraseña del servidor para establecer la comunicación, dicha configuración se presenta en el anexo 18.

Posteriormente se creó los ítems que se vinculan a los Things mediante enlaces llamados canales, y sus estados en la interfaz del usuario que se presenta en la figura 4.6, la creación de items se observa en la figura 4.7.

Figura 4.6 Items en openHAB



Items configurados openHAB. Elaborado por: Chinachi Andrés y Rios Jorge.

Figura 4.7 Archivo de configuración para Items

```

Number CisternaNivelCisterna "Nivel [%1f %%]" <cistern> {channel="mqtt:topic:cisterna:NivelCisterna"}
String SaloncomunalSensorHumoPB "Estado" <smoke> {channel="mqtt:topic:pbsmoke1:SensorHumoPB"}
Number SaloncomunalSensorHumoPBppm "Concentración [%1f ppm]" <smoke> {channel="mqtt:topic:pbsmoke1:SensorHumoPB_Ppm"}
Number GimnasioGenericMQTTThingTemperature "Temperatura del lugar [%1f °c]" <temperature> {channel="mqtt:topic:miflora:GenericMQTTThingTemperature"}
Number GimnasioGenericMQTTThingHumidity "Humedad [%1f %%]" <humidity> {channel="mqtt:topic:miflora:GenericMQTTThingHumidity"}
String ConserjeSensorGLPPB "Estado" <gas> {channel="mqtt:topic:pbglp1:SensorGLPPB"}
Number ConserjeSensorGLPPBppm "Concentración [%1f ppm]" <gas> {channel="mqtt:topic:pbglp1:SensorGLPPB_Ppm"}
String LavanderiaSensorFirePB "Estado" <fire> {channel="mqtt:topic:pbfire1:SensorFirePB"}
Switch Ventilacion_Edificio "Ventilación" <switch> {channel="mqtt:topic:miflora:Ventilacion"}

```

Ejemplo de configuración del archivo ítems en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

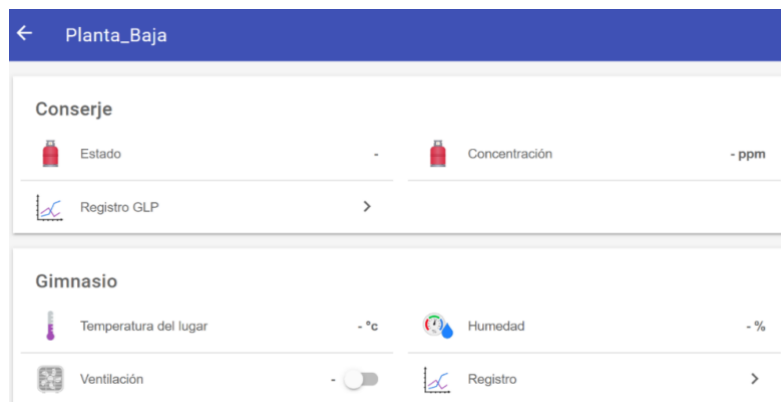
Se configuró el archivo de sitemaps desde Visual Studio, este se puede ver en la figura 4.8, dicho archivo permite presentar los ítems y things anteriormente creados, además, de creación de gráficas que se mostrarán remotamente, en una interfaz gráfica organizada mostrada en la figura 4.9.

Figura 4.8 Archivo de configuración para Sitemap

```
sitemap home label="Edificio Saint Chärbel"
{
  Frame{
    Group item=Planta_Baja
    {
      Frame label="Conserje"{
        Text item=ConserjeSensorGLPPB icon="gas"
        Text item=ConserjeSensorGLPPBPpm icon="gas"
        Group item=Registro_GLP label="Registro GLP" icon="line" {
          Chart item=ConserjeSensorGLPPBPpm label="Gráfico GLP Ppm vs tiempo" refresh=6000 period=12h ser
vice="influxdb" legend=true
        }
      }
    }
  }
}
```

Ejemplo de configuración del archivo sitemap en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

Figura 4.9 Interfaz Sitemap



Interfaz Sitemap configurada. Elaborado por: Chinachi Andrés y Rios Jorge.

Una vez establecido los requerimientos anteriores, se configuró el archivo rules para realizar la automatización de los dispositivos y servicios que se ofrecen en openHAB. En este archivo se configura las reglas y las condiciones en las que se realizarán diferentes acciones como el envío de notificaciones a los dispositivos conectados remotamente mediante la aplicación para celulares OpenHAB y mediante la plataforma Telegram y Whatsapp, dicho archivo se presenta en la figura 4.10.

Figura 4.10 Archivo de configuración para Rules

```
rule "Notification Nivel de agua cisterna"
when
  Item CisternaNivelCisterna changed
then
  if (CisternaNivelCisterna.state <= 30 && previousState >30 )
  {
    logInfo("Sensor Nivel Cisterna Subsuelo", "Nivel Bajo")
    sendBroadcastNotification("Bajo nivel en tanque+" + CisternaNivelCisterna.state.format("%.1f") + " %")
    sendTelegram("Administrador_SCH", "BAJO NIVEL DE AGUA EN LA CISTERNA")
    sendHttpGetRequest("https://api.callmebot.com/whatsapp.php?source=OpenHAB&phone=+593999797890&apikey=657312&text=Bajo+Nivel+en+la+Cisterna")
  }
end
```

Configuración del archivo rules en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.



Para el envío de notificaciones se usa el enlace con Telegram que permite vincular a OpenHAB haciendo uso de la API Telegram Bot mediante la creación de un Bot y vincularlo al servicio Telegram con el addon “Telegram Action” de OpenHAB para emitir y recibir mensajes, de igual forma se utiliza la API CallMeBot con la creación de un bot vinculado al servicio de Whatsapp, ambos se configuraron en el archivo rules, como se puede ver en el anexo 19. Los cambios realizados en los archivos de configuración se visualizarán en la interfaz Paper UI y Basic UI de OpenHAB

- Integración InfluxDB

Se instaló el addon “InfluxDB (v 1.0) Persistence” y en el archivo services del mismo se ingresa la información de configuración indicada en el anexo 20, para establecer como almacenamiento de información a la base de datos creada y llamada “openhab\_db”, que se observa en la figura 4.11.

Figura 4.11 Base de datos openhab\_db

```
andresdatabase@andresdatabase-VirtualBox:~$ influx -username admin -password ad
min
Connected to http://localhost:8086 version 1.8.1
InfluxDB shell version: 1.8.1
> show databases
name: databases
name
----
_internal
openhab_db
> use openhab_db
Using database openhab_db
> show measurements
name: measurements
name
----
CisternaNivelCisterna
ConserjeSensorGLPPB
ConserjeSensorGLPPBppm
GimnasioGenericMQTTThingHumidity
GimnasioGenericMQTTThingTemperature
LavanderiaSensorFirePB
SaloncomunalSensorHumoPB
SaloncomunalSensorHumoPBppm
```

Base de datos creada en influxDB. Elaborado por: Chinachi Andrés y Rios Jorge.

Para tener una gestión de gráficas se integra la plataforma Grafana para permitir la visualización de gráficas en la plataforma, se habilitó los servicios de autenticación anónima en el archivo de configuración de Grafana. Donde se crearon diferentes paneles para la visualización de los datos proporcionados por Grafana mediante la interfaz HABPanel, los paneles creados se pueden observar en el anexo 21. El HABPanel se ejecuta de manera local en un ordenador con conexión a internet, y las gráficas realizadas en Grafana, solo se visualizan en la máquina de administración dentro de la red local de la edificación.

#### **4.2.2.4. Integración Cloud**

Para extender la funcionalidad de la plataforma y permitir un monitoreo remoto de cada uno de los dispositivos en las instalaciones de la edificación sin la necesidad de estar en la red de la misma, se configuró una instancia en la nube basada en la comunidad OpenHab denominada myopenhab, que se consideró la más accesible ya que no se requiere ningún recurso físico y es completamente gratuita, permite un acceso remoto basado en un sistema de autenticación de usuarios que mediante una contraseña pueden acceder a las diferentes interfaces de la plataforma, ya sea desde un dispositivo móvil o desde un computador.

En myopenhab se configuró dos tipos de usuario que son maestro y regular, diferenciando que el usuario regular no puede agregar ni quitar usuarios teniendo solo acceso a la plataforma, además, la nube permite una mayor integración con la aplicación oficial de openhab al activar notificaciones push en los dispositivos móviles y wearables compatibles, accediendo a notificaciones en forma de alerta en cualquier lugar siempre que se tenga una conexión a internet. Para que el servicio cloud sea activado se registró en la página web [myopenhab.org](http://myopenhab.org) e introdujo los datos del administrador y las credenciales UUID y contraseña de la plataforma, que se obtienen en los archivos uuid y secret en la carpeta de openhab2 y openhab cloud respectivamente, los mismos fueron creados automáticamente una vez se ha instalado en addons de la plataforma el “openhab cloud conector”.

### **4.3. Pruebas y Resultados**

#### **4.3.1. Funcionamiento de la Plataforma IoT**

El funcionamiento del servidor OpenHAB, permitió la recopilación de los datos de los diferentes nodos sensores y la obtención de información del estado de los mismos mediante una interfaz gráfica diseñada para la facilidad de los usuarios. Dependiendo las condiciones en la que fueron programados los sensores descritos anteriormente, estos envían diferentes notificaciones vinculadas a cada nodo sensor al celular y a los usuarios que se encuentran enlazados a la plataforma mediante la integración cloud y a su vez mediante el accionamiento de una alarma sonora que permite el aviso de que una situación fuera de lo normal está sucediendo.

### 4.3.2. Lectura de Sensores

En la maqueta los sensores fueron distribuidos de forma tal que un sensor de temperatura cubre el gimnasio, un sensor de humo que cubre el salón comunal, el sensor de gas cubre el cuarto del conserje, el sensor de llama cubre la lavandería y se coloca un sensor de nivel en un tanque simulando la cisterna, la distribución de dichos sensores, se puede observar en la figura 4.12 y 4.13.

Figura 4.12 Maqueta realizada



Prototipo de planta baja del Edificio Saint Charbel. Elaborado por: Chinachi Andrés y Rios Jorge.

Figura 4.13 Prototipo de cisterna



Simulación de cisterna. Elaborado por: Chinachi Andrés y Rios Jorge.

Se planteó escenarios para verificar el funcionamiento de los sensores y su interacción con los servidores instalados, las pruebas básicas en cada uno de los sectores consisten en:

- **Alertas:** Los sensores a excepción del de temperatura, poseen buzzers que según el valor de alerta configurado son activados emitiendo un sonido y un destello de led indicador, los cuales, no se detendrán hasta que la situación esté en los valores normales nuevamente, estos valores se pueden ver mediante la representación gráficas de datos en el tiempo.
- **Notificaciones:** Haciendo uso de la plataforma cloud [myopenhab.org](http://myopenhab.org), mediante programación, se configuró el envío de notificaciones push que será recibido en todos los dispositivos inteligentes que las permitan, siempre y cuando se tenga configurado un usuario registrado en la plataforma, además, del envío de notificaciones a la plataforma Telegram y Whatsapp.
- **Gimnasio:** En las pruebas realizadas para el nodo sensor de temperatura se establece que al detectar una temperatura superior a 28 °C este procederá a la activación automática de un ventilador que representa un sistema de ventilación para dicha área, el cual a su vez está representado de manera gráfica en el sitemap con un botón tipo switch que permite observar que el ventilador se ha encendido, además este botón tiene la capacidad de ser accionado voluntariamente por el administrador de la plataforma en cualquier momento que se desee.
- **Salón Comunal:** El sensor de humo trabaja con el nivel de concentración de partículas según el gas que se desea medir, previamente ha sido escalado para detectar la concentración de humo y se ha calibrado para que se active cuando la concentración es considerablemente alta, para producir una concentración de humo densa, para probar que el sensor esté funcionando de manera óptima, se somete al sensor a diferentes concentraciones de humo, una vez que detecta la presencia de humo por encima de 400 ppm (partes por millón) de igual manera se activa una alarma sonora y simultáneamente envía una notificación a los dispositivos conectados al acceso remoto.
- **Conserje:** El sensor de gas trabaja de forma similar al sensor de humo, por tal ha sido escalado y calibrado para detectar una considerable concentración de presencia de GLP doméstico en el aire, la concentración es producida dejando

escapar gas doméstico hasta registrar valores por encima de 800 ppm, necesario para la activación de la alarma y notificación.

- Lavandería: El sensor de llama funciona detectando el espectro electromagnético al cual se encuentra el fuego, por tal la simulación de un incendio, se realiza mediante la activación de un encendedor produciendo fuego al cual el sensor responde rápidamente debido a su sensibilidad.
- Cisterna: En el sensor ultrasónico se simuló una disminución del tanque de la cisterna teniendo valores por debajo del 40%, una vez detectado se envía una notificación a los dispositivos conectados a la cuenta creada en myopenhab.org mediante la aplicación OpenHAB para celulares, Telegram y Whatsapp, a la vez que todas estas quedan registradas en la plataforma openhab cloud que se esté ejecutando en un ordenador.

En las tablas 4.1, 4.2, 4.3, 4.4 y 4.5, se observa los datos enviados de cada uno de los sensores al producirse un escenario fuera de lo normal, en sus respectivas áreas a la plataforma y su interacción con las interfaces Basic UI, Sitemaps y HABpanel, así como el envío de notificaciones, en consecuencia, a la correcta comunicación entre los diversos sistemas que permiten la operación y monitoreo de los sensores.

Tabla 4.1 Resultados sector Lavandería

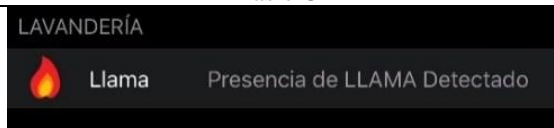


Área	Sensor	Interfaz
Lavandería	Fuego (KY-026)	Basic UI
		
		HABPANEL
		
		Notificación
		

Tabla de registro de alertas del sensor de fuego. Elaborado por: Chinachi Andrés y Rios Jorge.

Tabla 4.2 Resultados sector Gimnasio

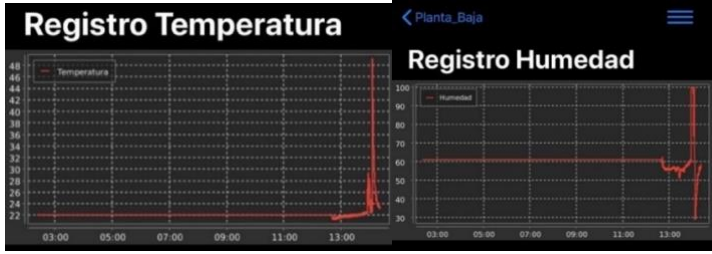


Área	Sensor	Interfaz
Gimnasio	Temperatura y Humedad (DHT22)	Basic UI
		
		HABPANEL
		
		Notificación
		

Tabla de registro de alertas sensor de temperatura. Elaborado por: Chinachi Andrés y Rios Jorge.

Tabla 4.3 Resultados Cisterna

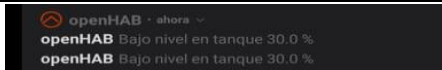
Área	Sensor	Interfaz
Cisterna	Nivel (HC-SR04)	Basic UI
		
		HABPANEL
		
		Notificación
		

Tabla de registro de alertas del sensor de nivel. Elaborado por: Chinachi Andrés y Rios Jorge.

Tabla 4.4 Resultados sector Conserje



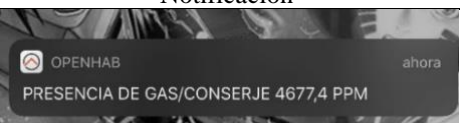
Área	Sensor	Interfaz
Conserje	Gas (MQ6)	Basic UI
		
		HABPANEL
		
		Notificación
		

Tabla de registro de alertas del sensor de gas GLP. Elaborado por: Chinachi Andrés y Rios Jorge.

Tabla 4.5 Resultados sector Salón Comunal


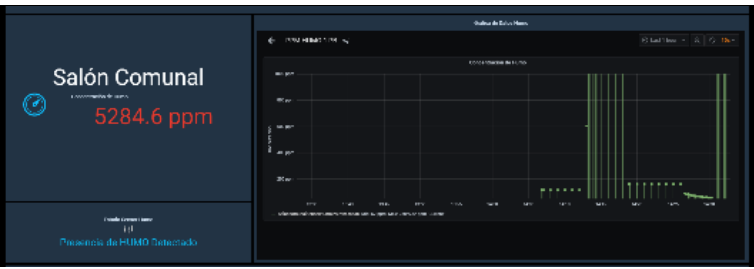
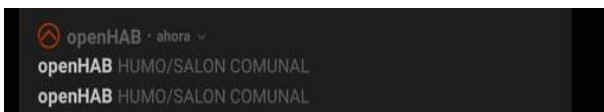
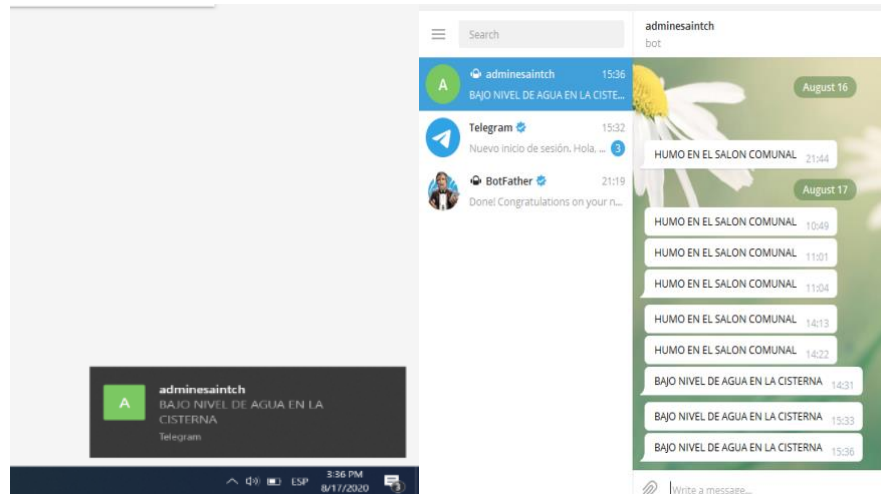
Área	Sensor	Interfaz
Salón Comunal	Humo (MQ2)	Basic UI
		
		HABPANEL
		
		Notificación
		

Tabla de registro de alertas del sensor de humo. Elaborado por: Chinachi Andrés y Rios Jorge.

Las notificaciones que llegan a Telegram se pueden observar en la figura 4.13, las cuales son mensajes enviados desde el Bot creado con la información establecida anteriormente.

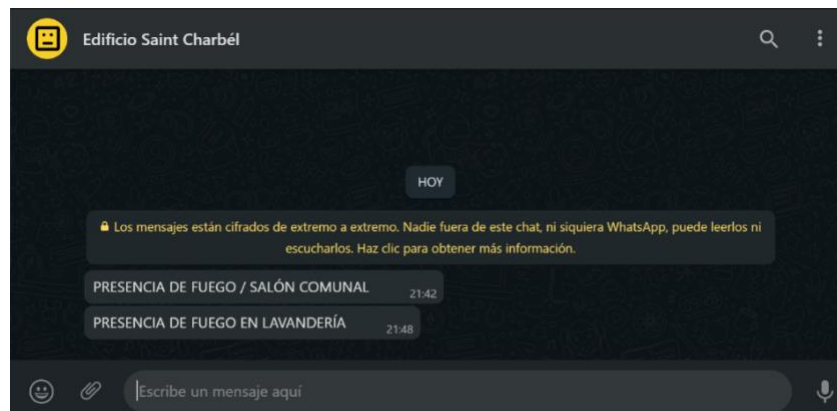
Figura 4.14 Notificaciones en Telegram



Chat de notificación con bot creado en Telegram. Elaborado por: Chinachi Andrés y Rios Jorge.

Las notificaciones en Whatsapp se observan en la figura 4.14, las cuales son mensajes configurados y enviados desde el Bot.

Figura 4.15 Notificaciones en Whatsapp



Chat de notificación con bot creado en Whatsapp. Elaborado por: Chinachi Andrés y Rios Jorge.



## CONCLUSIONES

La integración de componentes WSN e IoT en el edificio Saint Charbel, cumple con los diferentes estándares de operación y garantiza una baja latencia y administración remota en los dispositivos sensores y actuadores, además de la inclusión de una plataforma que permite un fácil monitoreo por parte de los empleados de la administración.

La plataforma OpenHab permite el control de dispositivos IoT multiplataforma, y brinda mayor flexibilidad en la integración de interfaces y servicios, con cualquier dispositivo sensor o actuador inalámbrico, que actúe con plataformas libres como Arduino y la mayoría de marcas existentes en el mercado.

La red WLAN diseñada permite la interconexión de los componentes WSN e IoT mediante tecnología Wifi de manera eficiente y óptima basándose en una topología en estrella que brinda una administración centralizada y escalabilidad en sus componentes y tecnologías.

En base a las pruebas de funcionamiento de la Red WSN y su gestión en la nube a través de la plataforma IoT OpenHab, se corrobora que el diseño propuesto implica una solución novedosa, eficiente y segura a bajo costo.

El prototipo implementado, permite corroborar el funcionamiento de la solución a través de la gestión de los dispositivos IoT y las diferentes formas de envío de notificaciones mediante la plataforma OpenHAB, mismas que se realizan en tiempo real, y permiten su representación gráfica que facilita su gestión.

El análisis de factibilidad económica determina una viabilidad positiva en la implementación del proyecto con una inversión de 40.986,26\$ y un TIR de 11%, garantizando una propuesta que da fuerza a la innovación en el edificio Saint Charbel y busca estar presente en la construcción de edificaciones residenciales.

## RECOMENDACIONES

Considerando la permanente funcionalidad que deben tener los sensores, se recomienda el uso de un sistema de alimentación secundario de tipo dedicado que permita la operación ininterrumpida de los actuadores y de los sistemas de seguridad.

Un complemento importante para la solución planteada sería la inclusión de políticas de seguridad tanto para los equipos físicos como las plataformas de control de las redes WLAN, WSN e IoT debido a que en su mayoría los elementos que estarán expuestos a los usuarios del edificio.

Para evitar accesos desautorizados y/o ataques de suplantación de identidad, se sugiere habilitar niveles de seguridad para el login de los diferentes sensores ya que no cuentan con una certificación de autenticidad tanto en suscripciones al servidor MQTT como a la plataforma IoT.

Debido a que la información que gestiona la red WSN es sensible, sería importante considerar el uso de servidores físicos tanto para la base de datos como para la plataforma IoT y el servidor MQTT, para garantizar una adecuada velocidad de transmisión, recepción y procesamiento de datos que demandan estas tecnologías y los equipos inherentes a la misma.

## BIBLIOGRAFÍA

- Aeromodelismo. (31 de Mayo de 2020). *Tipos de baterías para RadioControl*. Obtenido de R.C. TECNIC:  
[https://www.rctecnic.com/blog/12\\_Tipos-de-bater%C3%ADas-para-RadioControl.html](https://www.rctecnic.com/blog/12_Tipos-de-bater%C3%ADas-para-RadioControl.html)
- Aquino, E. A., Pérez Arriaga, J. C., Contreras Vega, G., & Cruz Juárez, A. (2016). Hacia un Middleware de Soporte para la Creación de Aplicaciones Accesibles a Personas con Discapacidad Visual dentro de una Red de Sensores. *Ática 2016. Tecnología y accesibilidad*, 176-183.
- Aranda, D. (2014). *Electrónica: Plataformas Arduino y raspberry pi*. Buenos Aires: Fox Andina.
- Bastidas Campaña, S. E., Meza, H. E., & Bastidas Campaña, A. J. (2018). *Capítulo 1: Las redes de sensores inalámbricas, arquitectura y aplicaciones*. Libros Universidad Nacional Abierta ya Distancia.
- Bensalama, M., Batatia, H., & Boucenna, M. L. (2015). *Ad Hoc Networks Telecommunications and Game Theory*. Hoboken: John Wiley & Sons, Incorporated.
- Bill Menezes, C. C. (24 de September de 2019). Magic Quadrant for Wired and Wireless LAN Access Infrastructure. *Gartner*.
- Calvo García, Á. L. (2014). *Gestión de redes telemáticas (UF1880)*. Andalucía: IC Editorial.
- Chain, N. S., Chain, R. S., & Puelma, J. S. (2014). *Preparación y evaluación de proyectos* (Vol. VI). México, Colombia: McGraw-Hill.
- Constructora SAUD. (2019). *Saud Constructora*. Recuperado el Julio de 2020, de <http://www.constructorasaud.com/quienes>
- Cruz, M. V. (2015). *Las tecnologías IoT dentro de la Industria conectada 4.0* (Vol. Primera). Madrid, España: Fundación EOI.
- David, H. G. (2017). *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things* (Vol. First). Indianapolis, USA: Cisco Press.
- Dehmer, M., Streib, F. E., & Pickl, S. (2015). *Computational Network Theory : Theoretical Foundations and Applications*. weinheim: John Wiley & Sons, Incorporated.

domoticaintegrada. (14 de Agosto de 2018). *¿Cómo funciona y cuál es el uso de un sensor de agua?* Obtenido de Domótica Integrada:

[https://domoticaintegrada.com/sensor-de-agua/#Sensor\\_de\\_agua\\_por\\_ultrasonidos](https://domoticaintegrada.com/sensor-de-agua/#Sensor_de_agua_por_ultrasonidos)

Goasduff, L. (23 de July de 2019). *Gartner Identifies the Top 10 Wireless Technology Trends for 2019 and Beyond*. Recuperado el Mayo de 2020, de Gartner: <https://www.gartner.com/en/newsroom/press-releases/2019-07-23-gartner-identifies-the-top-10-wireless-technology-tre>

Guzmán Navarro, F., & Merino Córdoba, S. (2015). *Domótica: gestión de la energía y gestión técnica de edificios*. Madrid: RA-MA Editorial.

Hérault, L., Seghrouchni, A. E., Ishikawa, F., & Tokuda, H. (2016). *Enablers for Smart Cities*. Hoboken: John Wiley & Sons, Incorporated.

Hosch, M. (1 de Abril de 2019). *Meeting NFPA standards for flame, smoke and gas detection*. Obtenido de Gulf Fire: <https://gulffire.mdmpublishing.com/meeting-nfpa-standards-for-flame-smoke-and-gas-detection/>

Huidobro, J. M. (2014). *Telecomunicaciones: tecnologías, redes y servicios (2a. ed.)*. Madrid: RA-MA Editorial.

Le, D.-N. B. (2019). *Security Designs for the Cloud, Iot, and Social Networking (Vol. First)*. Hoboken, USA: Scrivener Publishing LLC.

Lea, P. (2018). *Internet of Things for Architects Architecting IoT solutions by implementing sensors, communication infrastructure, edge computing, analytics, and security (Vol. First)*. Birmingham, Mumbai: Packt Publishing Ltd.

Leviton. (05 de Mayo de 2020). *sensores de ocupación*. Obtenido de grupoeib: [http://grupoeib.com/index.php?controller=attachment&id\\_attachment=28](http://grupoeib.com/index.php?controller=attachment&id_attachment=28)

Llaneza, P. (2018). *Seguridad y Responsabilidad en la Internet de las Cosas (IoT) (Vol. Primera)*. Madrid, España: Wolters Kluwer España, S.A.

Molina Robles, F. J. (2014). *Implantación de los elementos de la red local*. Madrid: RA-MA Editorial.

National Fire Protection Assosiation. (27 de Mayo de 2020). *Installing and maintaining smoke alarms - NFPA*. Obtenido de National Fire Protection Assosiation: <https://www.nfpa.org/Public-Education/Staying-safe/Safety-equipment/Smoke-alarms/Installing-and-maintaining-smoke-alarms>

Ordóñez, J. L., & Huidobro, J. M. (2014). *Comunicaciones por radio: tecnologías, redes y servicios de radiocomunicaciones: el espectro electromagnético* (2 ed.). Madrid: RA-MA Editorial.

Peng, S.-I. P. (2020). *Principles of Internet of Things (IoT) Ecosystem: Insight Paradigm* (Vol. Vol. 174). Sydney, Australia: Springer Nature Switzerland AG.

Robles, F. M. (2016). *Planificación y administración de redes*. Madrid: RA-MA Editorial.

rodychadmin. (8 de Noviembre de 2016). *DETECTORES DE GAS. ¿DONDE COLOCARLOS?* Obtenido de Rodych Seguridad: <https://rodych.es/detectores-de-gas-donde-colocarlos/>

Romero Amondaray, L., Piña, F. M., & Goire Castilla, M. M. (2018). Red de sensores inalámbricos para las casas de cultivos protegidos "San Jose"/Wireless Sensor Network for "San Jose" greenhouse. *Gale Academic*, 39, 16+.

Rueda, J. S., & Talavera, J. M. (2017). Similitudes y diferencias entre Redes de Sensores Inalámbricas e Internet de las Cosas: Hacia una postura clarificadora. *Revista Colombiana de computación*, 58-74.

Serpanos, D. W. (2018). *Internet-of-Things ( IoT ) Systems Architectures, Algorithms, Methodologies* (Vol. First). Atlanta, USA: Springer International Publishing.

Slama, D. P. (2015). *Enterprise IoT Strategies & Best Practices for Connected Products & Services* (Vol. Firts). United States of America: O'Reilly Media, Inc.,.

Song, H., Srinivasan, R., Sookoor, T., Jeschke, S., & Jeschke, S. (2017). *Smart Cities : Foundations, Principles, and Applications*. Hoboken: John Wiley & Sons, Incorporated.

Urbina, G. B. (2016). *Evaluación de proyectos* (Vol. VIII). Ciudad de México: McGRAW-HILL.

wizlynx group. (8 de Jullio de 2020). *Servicios de Evaluación de la Seguridad de los Sistemas de Control Industrial (SIC)*. Obtenido de [wizlynxgroup.com: https://www.wizlynxgroup.com/mx/ciberseguridad-mexico/evaluacion-de-la-seguridad-de-los-sistemas-de-control-industrial](https://www.wizlynxgroup.com/mx/ciberseguridad-mexico/evaluacion-de-la-seguridad-de-los-sistemas-de-control-industrial)

## ANEXOS

### Anexo 1: Distribución de fibras de la red GPON

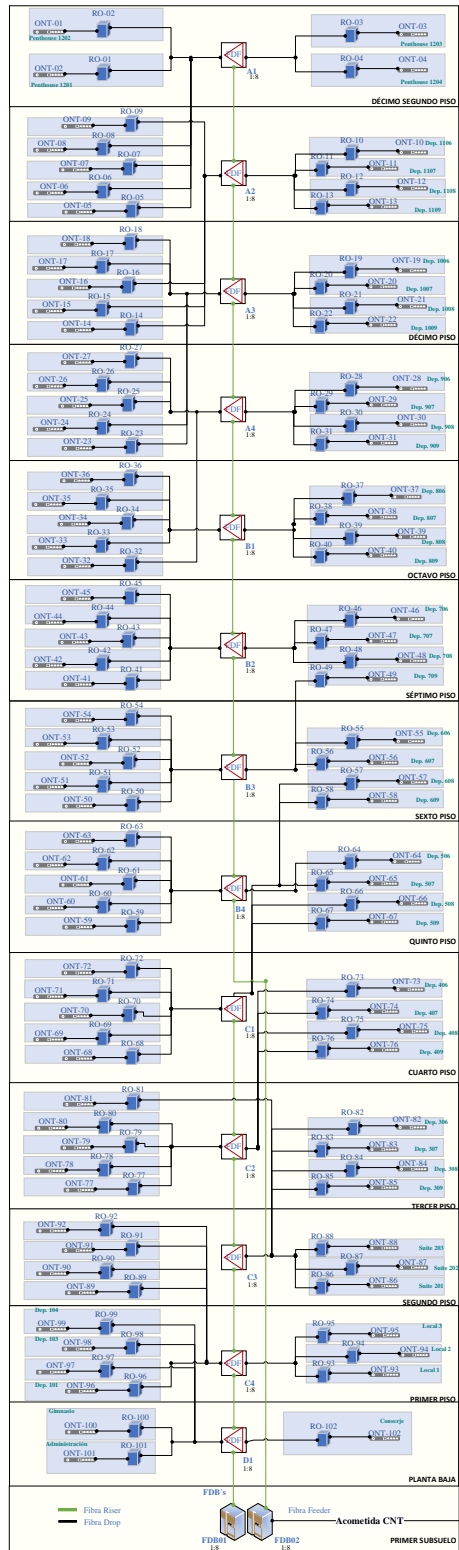
Distribución de fibras para abonados.

<b>Platas</b>	<b>Fibras Ópticas</b>
<b>Subsuelo</b>	
4	-
3	-
2	-
1	-
<b>Planta</b>	
Baja	3
<b>Pisos</b>	
1	7
2	7
3	9
4	9
5	9
6	9
7	9
8	9
9	9
10	9
11	9
12	4
<b>Total</b>	<b>102</b>

Distribución de fibras para abonados. Elaborado por: Chinachi Andrés, Rios Jorge.

## Anexo 2: Topología del Edificio de la red GPON

### Topología física del edificio



Topología de red física en la edificación. Elaborado por: Chinachi Andrés, Rios Jorge.

### Anexo 3: Distribución Lógica de la red GPON

#### Distribución Lógica de Fibras.

Manga	Hilos de Splitter en manga	Splitters FDB	FD Serie	Piso FDF	Hilos de FDF	Piso	Roseta Óptica	Ubicación
Splitter principal 1 SP-01 1:8	1	SS-01 1:8	A1	Piso 12	1	Piso 12	RO-01	Penthouse 1201
					2	Piso 12	RO-02	Penthouse 1202
					3	Piso 12	RO-03	Penthouse 1203
					4	Piso 12	RO-04	Penthouse 1204
					5	Piso 11	RO-05	Departamento 1101
					6	Piso 11	RO-06	Suite 1102
					7	Piso 11	RO-07	Suite 1103
					8	Piso 11	RO-08	Departamento 1104
	2	SS-02 1:8	A2	Piso 11	1	Piso 11	RO-09	Suite 1105
					2	Piso 11	RO-10	Departamento 1106
					3	Piso 11	RO-11	Departamento 1107
					4	Piso 11	RO-12	Departamento 1108
					5	Piso 11	RO-13	Departamento 1109
					6	Piso 10	RO-14	Departamento 1001
					7	Piso 10	RO-15	Suite 1002
					8	Piso 10	RO-16	Suite 1003
	3	SS-03 1:8	A3	Piso 10	1	Piso 10	RO-17	Departamento 1004
					2	Piso 10	RO-18	Suite 1005
					3	Piso 10	RO-19	Departamento 1006
					4	Piso 10	RO-20	Departamento 1007
					5	Piso 10	RO-21	Departamento 1008
					6	Piso 10	RO-22	Departamento 1009
					7	Piso 9	RO-23	Departamento 901



			8			8	Piso 9	RO-24	Suite 902
4	SS-04 1:8	A4	Piso 9	1	1	Piso 9	RO-25	Suite 903	
				2	2	Piso 9	RO-26	Departamento 904	
				3	3	Piso 9	RO-27	Suite 905	
				4	4	Piso 9	RO-28	Departamento 906	
				5	5	Piso 9	RO-29	Departamento 907	
				6	6	Piso 9	RO-30	Departamento 908	
				7	7	Piso 9	RO-31	Departamento 909	
				8	8	Piso 8	RO-32	Departamento 801	
5	SS-05 1:8	B1	Piso 8	1	1	Piso 8	RO-33	Suite 802	
				2	2	Piso 8	RO-34	Suite 803	
				3	3	Piso 8	RO-35	Departamento 804	
				4	4	Piso 8	RO-36	Suite 805	
				5	5	Piso 8	RO-37	Departamento 806	
				6	6	Piso 8	RO-38	Departamento 807	
				7	7	Piso 8	RO-39	Departamento 808	
				8	8	Piso 8	RO-40	Departamento 809	
6	SS-06 1:8	B2	Piso 7	1	1	Piso 7	RO-41	Departamento 701	
				2	2	Piso 7	RO-42	Suite 702	
				3	3	Piso 7	RO-43	Suite 703	
				4	4	Piso 7	RO-44	Departamento 704	
				5	5	Piso 7	RO-45	Suite 705	
				6	6	Piso 7	RO-46	Departamento 706	
				7	7	Piso 7	RO-47	Departamento 707	
				8	8	Piso 7	RO-48	Departamento 708	
7	SS-07 1:8	B3	Piso 6	1	1	Piso 7	RO-49	Departamento 709	
				2	2	Piso 6	RO-50	Departamento 601	
				3	3	Piso 6	RO-51	Suite 602	
				4	4	Piso 6	RO-52	Suite 603	
				5	5	Piso 6	RO-53	Departamento 604	
				6	6	Piso 6	RO-54	Suite 605	
				7	7	Piso 6	RO-55	Departamento 606	
				8	8	Piso 6	RO-56	Departamento 607	

	8	SS-08 1:8	1	B4	Piso 5	1	Piso 6	RO-57	Departamento 608
			2			2	Piso 6	RO-58	Departamento 609
			3			3	Piso 5	RO-59	Departamento 501
			4			4	Piso 5	RO-60	Suite 502
			5			5	Piso 5	RO-61	Suite 503
			6			6	Piso 5	RO-62	Departamento 504
			7			7	Piso 5	RO-63	Suite 505
			8			8	Piso 5	RO-64	Departamento 506
Splitter principal 2 SP-02 1:8	1	SS-09 1:8	1	C1	Piso 4	1	Piso 5	RO-65	Departamento 507
			2			2	Piso 5	RO-66	Departamento 508
			3			3	Piso 5	RO-67	Departamento 509
			4			4	Piso 4	RO-68	Departamento 401
			5			5	Piso 4	RO-69	Suite 402
			6			6	Piso 4	RO-70	Suite 403
			7			7	Piso 4	RO-71	Departamento 404
			8			8	Piso 4	RO-72	Suite 405
	2	SS-10 1:8	C2	Piso 3	1	Piso 4	RO-73	Departamento 406	
					2	2	Piso 4	RO-74	Departamento 407
					3	3	Piso 4	RO-75	Departamento 408
					4	4	Piso 4	RO-76	Departamento 409
					5	5	Piso 3	RO-77	Departamento 301
					6	6	Piso 3	RO-78	Suite 302
					7	7	Piso 3	RO-79	Suite 303
					8	8	Piso 3	RO-80	Departamento 304
	3	SS-11 1:8	C3	Piso 2	1	Piso 3	RO-81	Suite 305	
					2	2	Piso 3	RO-82	Departamento 306
					3	3	Piso 3	RO-83	Departamento 307
					4	4	Piso 3	RO-84	Departamento 308
					5	5	Piso 3	RO-85	Departamento 309
					6	6	Piso 2	RO-86	Departamento 201
					7	7	Piso 2	RO-87	Suite 202
					8	8	Piso 2	RO-88	Suite 203
4	SS-12	1	C4	Piso 1	1	Piso 2	RO-89	Departamento 204	

		1:8	2			2	Piso 2	RO-90	Suite 205
			3			3	Piso 2	RO-91	Departamento 206
			4			4	Piso 2	RO-92	Departamento 207
			5			5	Piso 1	RO-93	LOCAL 1
			6			6	Piso 1	RO-94	LOCAL 2
			7			7	Piso 1	RO-95	LOCAL 3
			8			8	Piso 1	RO-96	Departamento 101
			5			SS-13 1:8	D1	Planta Baja	1
	2	2		Piso 1	RO-98				Departamento 103
	3	3		Piso 1	RO-99				Departamento 104
	4	4		Planta Baja	RO-100				Administración
	5	5		Planta Baja	RO-101				Gimnasio
	6	6		Planta Baja	RO-102				Conserje
	7	7			RESERVA				
	8	8			RESERVA				

Tabla de distribución lógica de fibras ópticas en el edificio. Elaborado por: Chinachi Andrés, Ríos Jorge.

#### Anexo 4: Direccionamiento de la red GPON

##### Plan de direccionamiento

Dispositivo	Etiqueta	Planta	Etiqueta Roseta conexión	Dirección	Máscara
Optical Network Terminal	ONT-1	12	RO-01	172.20.0.0	24
Optical Network Terminal	ONT-2	12	RO-02	172.20.1.0	24
Optical Network Terminal	ONT-3	12	RO-03	172.20.2.0	24
Optical Network Terminal	ONT-4	12	RO-04	172.20.3.0	24
Optical Network Terminal	ONT-5	11	RO-05	172.20.4.0	26

Optical Network Terminal	ONT-6	11	RO-06	172.20.4.64	26
Optical Network Terminal	ONT-7	11	RO-07	172.20.4.128	26
Optical Network Terminal	ONT-8	11	RO-08	172.20.4.192	26
Optical Network Terminal	ONT-9	11	RO-09	172.20.5.0	26
Optical Network Terminal	ONT-10	11	RO-10	172.20.5.64	26
Optical Network Terminal	ONT-11	11	RO-11	172.20.5.128	26
Optical Network Terminal	ONT-12	11	RO-12	172.20.5.192	26
Optical Network Terminal	ONT-13	11	RO-13	172.20.6.0	26
Optical Network Terminal	ONT-14	10	RO-14	172.20.6.64	26
Optical Network Terminal	ONT-15	10	RO-15	172.20.6.128	26
Optical Network Terminal	ONT-16	10	RO-16	172.20.6.192	26
Optical Network Terminal	ONT-17	10	RO-17	172.20.7.0	26
Optical Network Terminal	ONT-18	10	RO-18	172.20.7.64	26
Optical Network Terminal	ONT-19	10	RO-19	172.20.7.128	26
Optical Network Terminal	ONT-20	10	RO-20	172.20.7.192	26
Optical Network Terminal	ONT-21	10	RO-21	172.20.8.0	26

Optical Network Terminal	ONT-22	10	RO-22	172.20.8.64	26
Optical Network Terminal	ONT-23	9	RO-23	172.20.8.128	26
Optical Network Terminal	ONT-24	9	RO-24	172.20.8.192	26
Optical Network Terminal	ONT-25	9	RO-25	172.20.9.0	26
Optical Network Terminal	ONT-26	9	RO-26	172.20.9.64	26
Optical Network Terminal	ONT-27	9	RO-27	172.20.9.128	26
Optical Network Terminal	ONT-28	9	RO-28	172.20.9.192	26
Optical Network Terminal	ONT-29	9	RO-29	172.20.10.0	26
Optical Network Terminal	ONT-30	9	RO-30	172.20.10.64	26
Optical Network Terminal	ONT-31	9	RO-31	172.20.10.128	26
Optical Network Terminal	ONT-32	8	RO-32	172.20.10.192	26
Optical Network Terminal	ONT-33	8	RO-33	172.20.11.0	26
Optical Network Terminal	ONT-34	8	RO-34	172.20.11.64	26
Optical Network Terminal	ONT-35	8	RO-35	172.20.11.128	26
Optical Network Terminal	ONT-36	8	RO-36	172.20.11.192	26
Optical Network Terminal	ONT-37	8	RO-37	172.20.12.0	26

Optical Network Terminal	ONT-38	8	RO-38	172.20.12.64	26
Optical Network Terminal	ONT-39	8	RO-39	172.20.12.128	26
Optical Network Terminal	ONT-40	8	RO-40	172.20.12.192	26
Optical Network Terminal	ONT-41	7	RO-41	172.20.13.0	26
Optical Network Terminal	ONT-42	7	RO-42	172.20.13.64	26
Optical Network Terminal	ONT-43	7	RO-43	172.20.13.128	26
Optical Network Terminal	ONT-44	7	RO-44	172.20.13.192	26
Optical Network Terminal	ONT-45	7	RO-45	172.20.14.0	26
Optical Network Terminal	ONT-46	7	RO-46	172.20.14.64	26
Optical Network Terminal	ONT-47	7	RO-47	172.20.14.128	26
Optical Network Terminal	ONT-48	7	RO-48	172.20.14.192	26
Optical Network Terminal	ONT-49	7	RO-49	172.20.15.0	26
Optical Network Terminal	ONT-50	6	RO-50	172.20.15.64	26
Optical Network Terminal	ONT-51	6	RO-51	172.20.15.128	26
Optical Network Terminal	ONT-52	6	RO-52	172.20.15.192	26
Optical Network Terminal	ONT-53	6	RO-53	172.20.16.0	26

Optical Network Terminal	ONT-54	6	RO-54	172.20.16.64	26
Optical Network Terminal	ONT-55	6	RO-55	172.20.16.128	26
Optical Network Terminal	ONT-56	6	RO-56	172.20.16.192	26
Optical Network Terminal	ONT-57	6	RO-57	172.20.17.0	26
Optical Network Terminal	ONT-58	6	RO-58	172.20.17.64	26
Optical Network Terminal	ONT-59	5	RO-59	172.20.17.128	26
Optical Network Terminal	ONT-60	5	RO-60	172.20.17.192	26
Optical Network Terminal	ONT-61	5	RO-61	172.20.18.0	26
Optical Network Terminal	ONT-62	5	RO-62	172.20.18.64	26
Optical Network Terminal	ONT-63	5	RO-63	172.20.18.128	26
Optical Network Terminal	ONT-64	5	RO-64	172.20.18.192	26
Optical Network Terminal	ONT-65	5	RO-65	172.20.19.0	26
Optical Network Terminal	ONT-66	5	RO-66	172.20.19.64	26
Optical Network Terminal	ONT-67	5	RO-67	172.20.19.128.0	26
Optical Network Terminal	ONT-68	4	RO-68	172.20.19.192	26
Optical Network Terminal	ONT-69	4	RO-69	172.20.20.0	26

Optical Network Terminal	ONT-70	4	RO-70	172.20.20.64	26
Optical Network Terminal	ONT-71	4	RO-71	172.20.20.128	26
Optical Network Terminal	ONT-72	4	RO-72	172.20.20.192	26
Optical Network Terminal	ONT-73	4	RO-73	172.20.21.0	26
Optical Network Terminal	ONT-74	4	RO-74	172.20.21.64	26
Optical Network Terminal	ONT-75	4	RO-75	172.20.21.128	26
Optical Network Terminal	ONT-76	4	RO-76	172.20.21.192	26
Optical Network Terminal	ONT-77	3	RO-77	172.20.22.0	26
Optical Network Terminal	ONT-78	3	RO-78	172.20.22.64	26
Optical Network Terminal	ONT-79	3	RO-79	172.20.22.128	26
Optical Network Terminal	ONT-80	3	RO-80	172.20.22.192	26
Optical Network Terminal	ONT-81	3	RO-81	172.20.23.0	26
Optical Network Terminal	ONT-82	3	RO-82	172.20.23.64	26
Optical Network Terminal	ONT-83	3	RO-83	172.20.23.128	26
Optical Network Terminal	ONT-84	3	RO-84	172.20.23.192	26
Optical Network Terminal	ONT-85	3	RO-85	172.20.24.0	26



Optical Network Terminal	ONT-86	2	RO-86	172.20.24.64	26
Optical Network Terminal	ONT-87	2	RO-87	172.20.24.128	26
Optical Network Terminal	ONT-88	2	RO-88	172.20.24.192	26
Optical Network Terminal	ONT-89	2	RO-89	172.20.25.0	26
Optical Network Terminal	ONT-90	2	RO-90	172.20.25.64	26
Optical Network Terminal	ONT-91	2	RO-91	172.20.25.128	26
Optical Network Terminal	ONT-92	2	RO-92	172.20.25.192	26
Optical Network Terminal	ONT-93	1	RO-93	172.20.26.0	26
Optical Network Terminal	ONT-94	1	RO-94	172.20.26.64	26
Optical Network Terminal	ONT-95	1	RO-95	172.20.26.128	26
Optical Network Terminal	ONT-96	1	RO-96	172.20.26.192	26
Optical Network Terminal	ONT-97	1	RO-97	172.20.27.0	26
Optical Network Terminal	ONT-98	1	RO-98	172.20.27.64	26
Optical Network Terminal	ONT-99	1	RO-99	172.20.27.128	26
Optical Network Terminal	ONT-100	Baja	RO-100	172.20.28.0	27
Optical Network Terminal	ONT-101	Baja	RO-101	172.20.27.192	26
Optical Network Terminal	ONT-102	Baja	RO-102	172.20.28.32	27

Tabla de plan de direccionamiento para el edificio. Elaborado por: Chinachi Andrés, Rios Jorge.

## Anexo 5: Distribución de APs.

### APs Subsuelo 4.



Distribución de APs en el Subsuelo 4. Elaborado por: Chinachi Andrés, Rios Jorge.

### APs Subsuelo 3.



Distribución de APs en el Subsuelo 3. Elaborado por: Chinachi Andrés, Rios Jorge.

### APS Subsuelo 1



Distribución de APs en el Subsuelo 1. Elaborado por: Chinachi Andrés, Rios Jorge.

### APs Piso 1.



Distribución de APs en el Piso 1. Elaborado por: Chinachi Andrés, Rios Jorge.

APs Piso 2.



Distribución de APs en el Piso 2. Elaborado por: Chinachi Andrés, Ríos Jorge.

APs Piso 3-11.



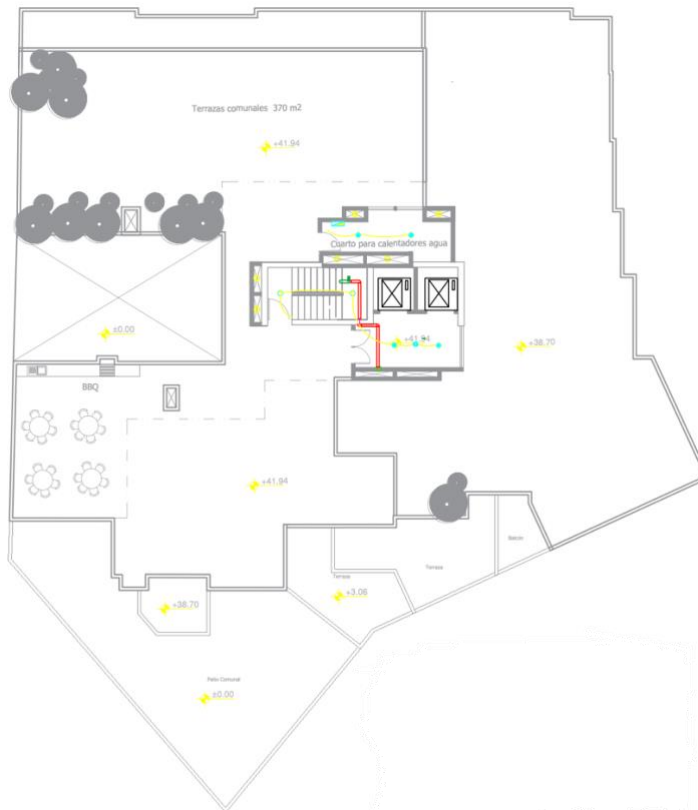
Distribución de cableado del Piso 3 al Piso 11. Elaborado por: Chinachi Andrés, Ríos Jorge.

### APs Piso 12.



Distribución de APs en el Piso 12. Elaborado por: Chinachi Andrés, Rios Jorge.

### APs Terraza.



Distribución de APs en la Terraza. Elaborado por: Chinachi Andrés, Rios Jorge.

## Anexo 6: Distribución de Sensores.

### Distribución Subsuelo 4.



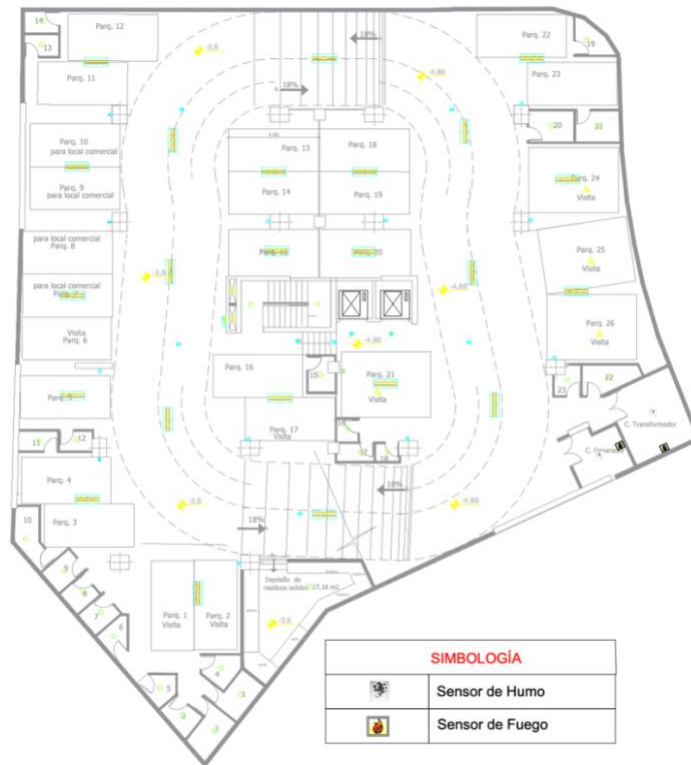
Distribución de sensores en el subsuelo 4. Elaborado por: Chinachi Andrés, Rios Jorge.

### Distribución Subsuelo 3.



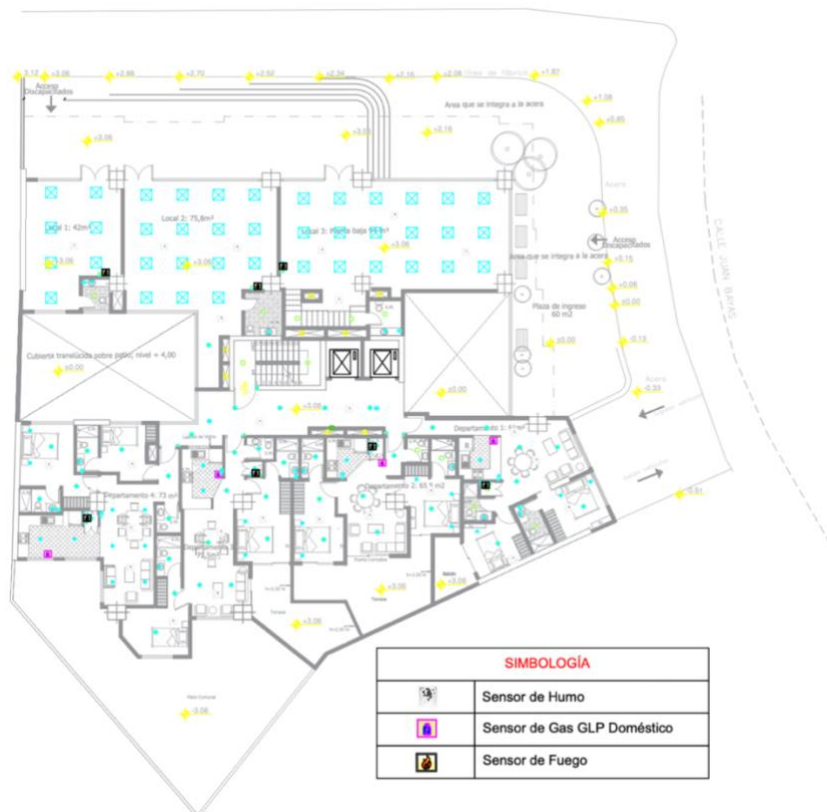
Distribución de sensores en el subsuelo 3. Elaborado por: Chinachi Andrés, Rios Jorge.

### Distribución Subsuelo 1.



Distribución de sensores en el subsuelo 1. Elaborado por: Chinachi Andrés, Rios Jorge.

### Distribución Piso 1.



Distribución de sensores en el Piso 1. Elaborado por: Chinachi Andrés, Rios Jorge.



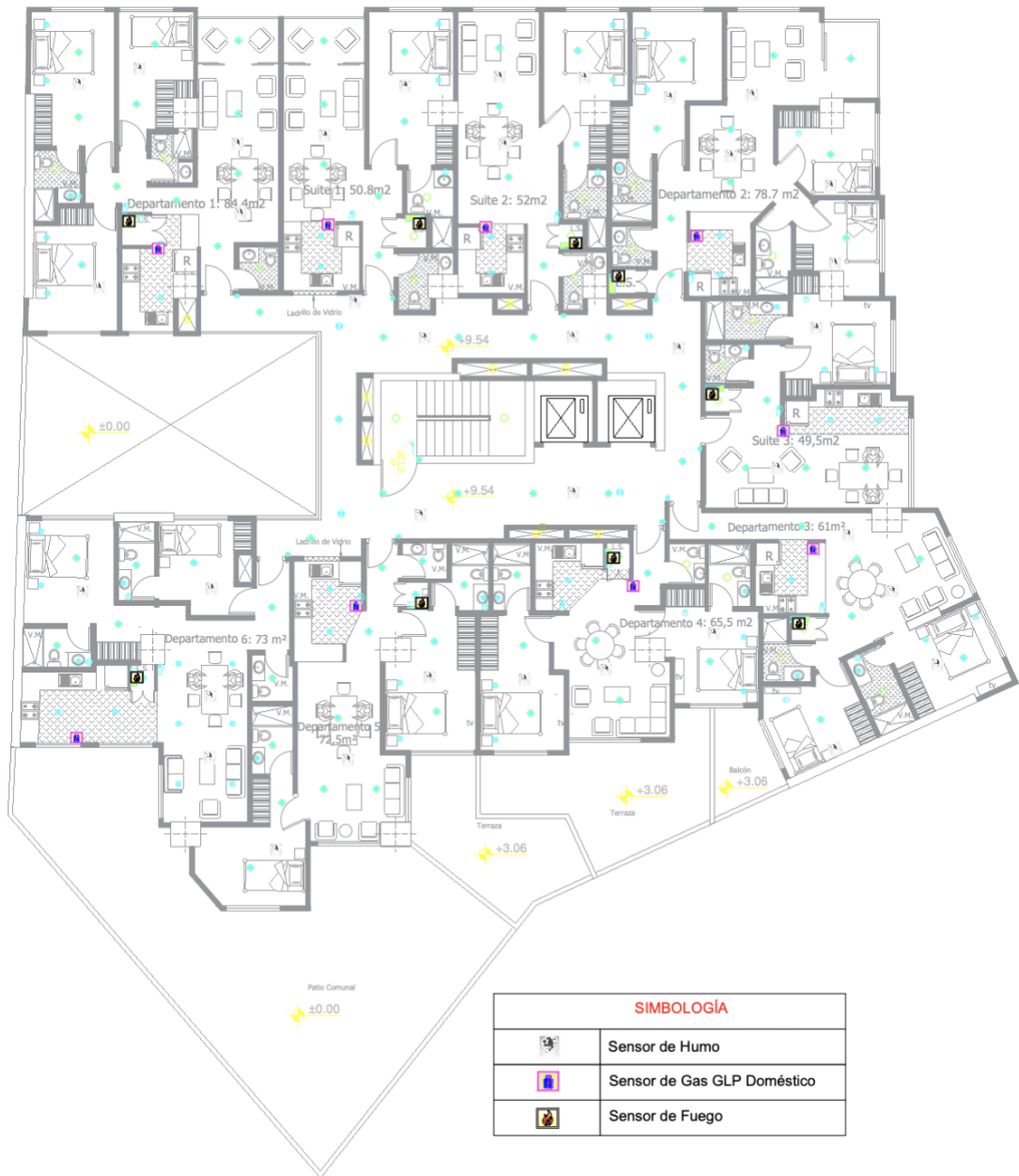
## Distribución Piso 2.



Distribución de sensores en el Piso 2. Elaborado por: Chinachi Andrés, Rios Jorge.

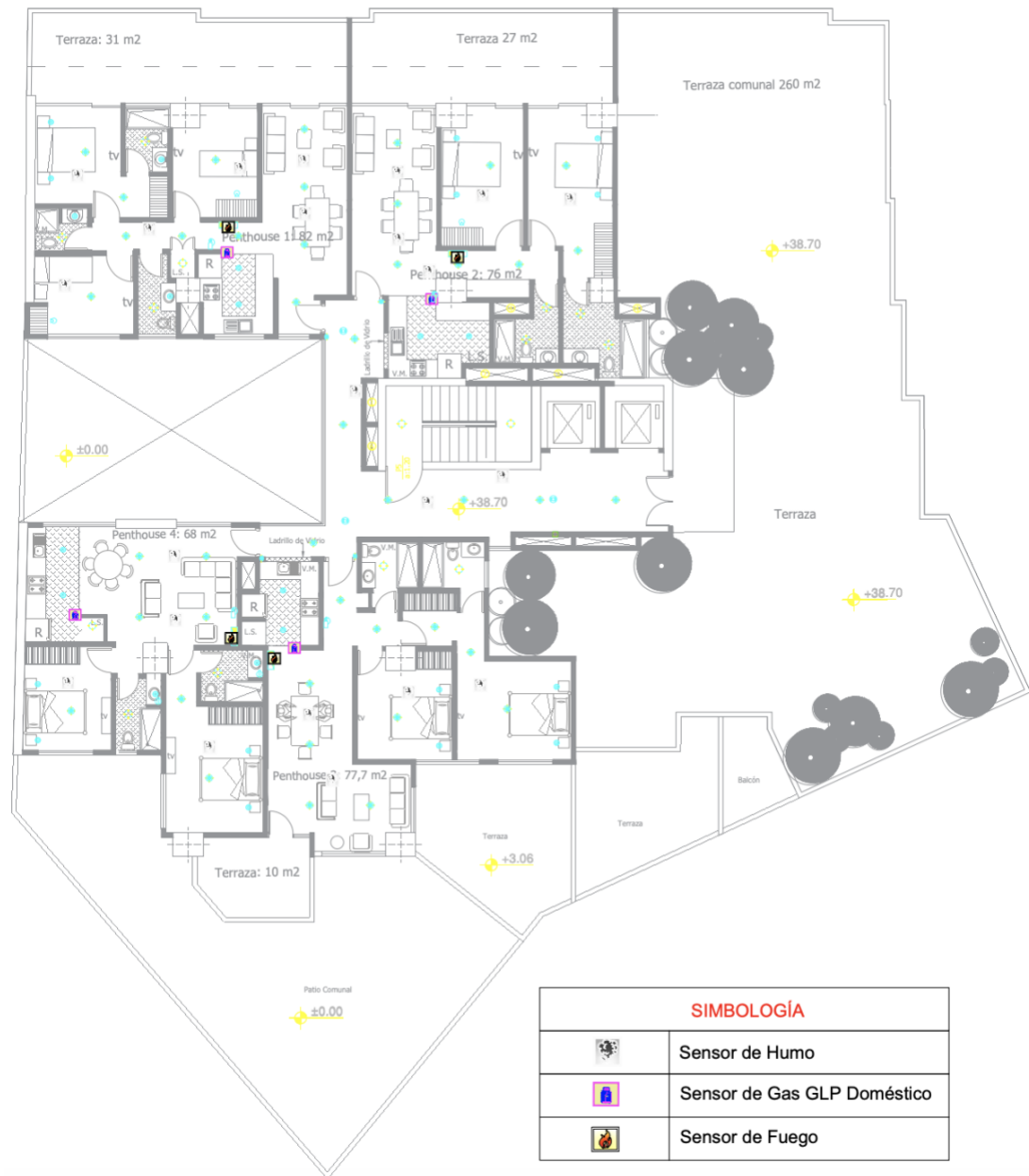


### Distribución Piso 3-11.



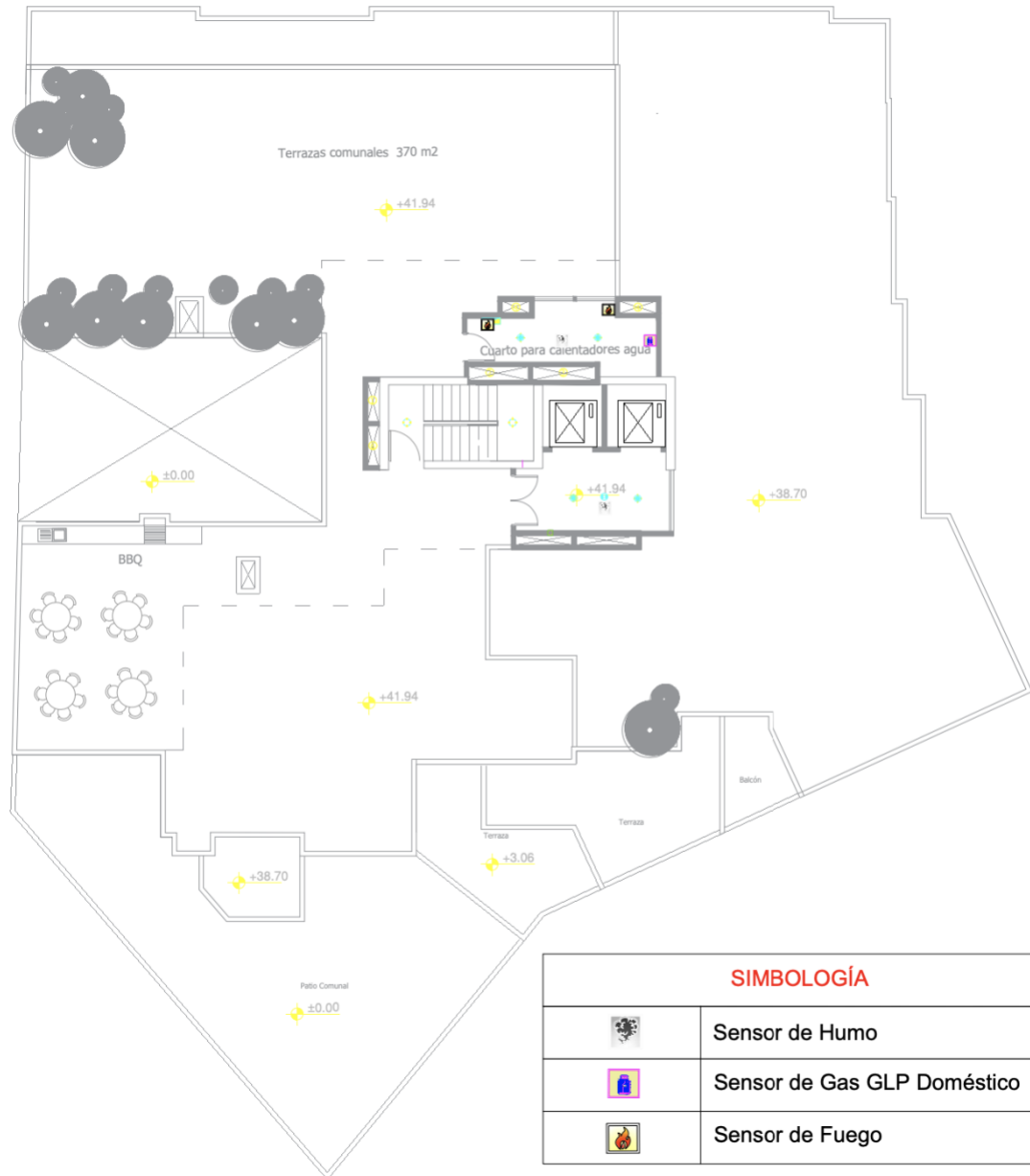
Distribución de sensores del Piso 3 al Piso 11. Elaborado por: Chinachi Andrés, Rios Jorge.

## Distribución Piso 12.



Distribución de sensores en el Piso12. Elaborado por: Chinachi Andrés, Rios Jorge.

## Distribución Terraza.



Distribución de sensores en la Terraza. Elaborado por: Chinachi Andrés, Rios Jorge.

**Anexo 7:** Selección y Análisis comparativo de topología de red.

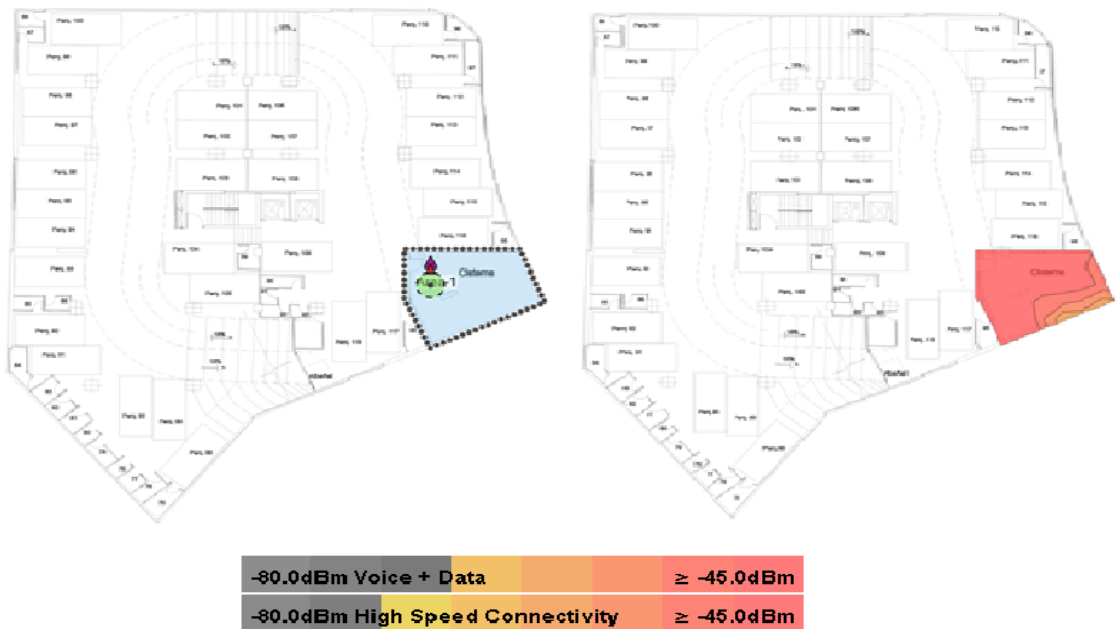
Topologías de la red.

Topologías Parámetros	Estrella extendida	Malla	Arbol
Permite centralizar puntos de fallo	X		
Menor Latencia	X		
Menor gasto energético	X		
Menos posibilidad de colisiones	X		
Cubre mayor distancia		X	X
Mayor tolerancia a fallos		X	X
Selección			

Tabla de selección y comparación de topologías de red. laborado por: Chinachi Andrés, Rios Jorge.

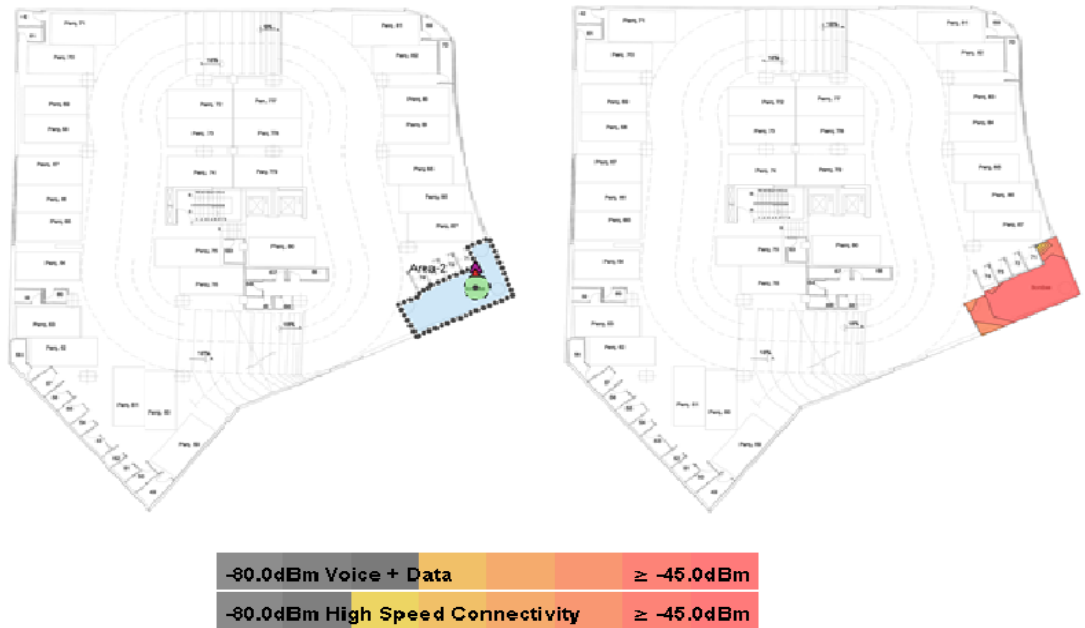
**Anexo 8:** Mapas de calor (site survey).

Mapa de intensidad de señal para el Subsuelo 4.



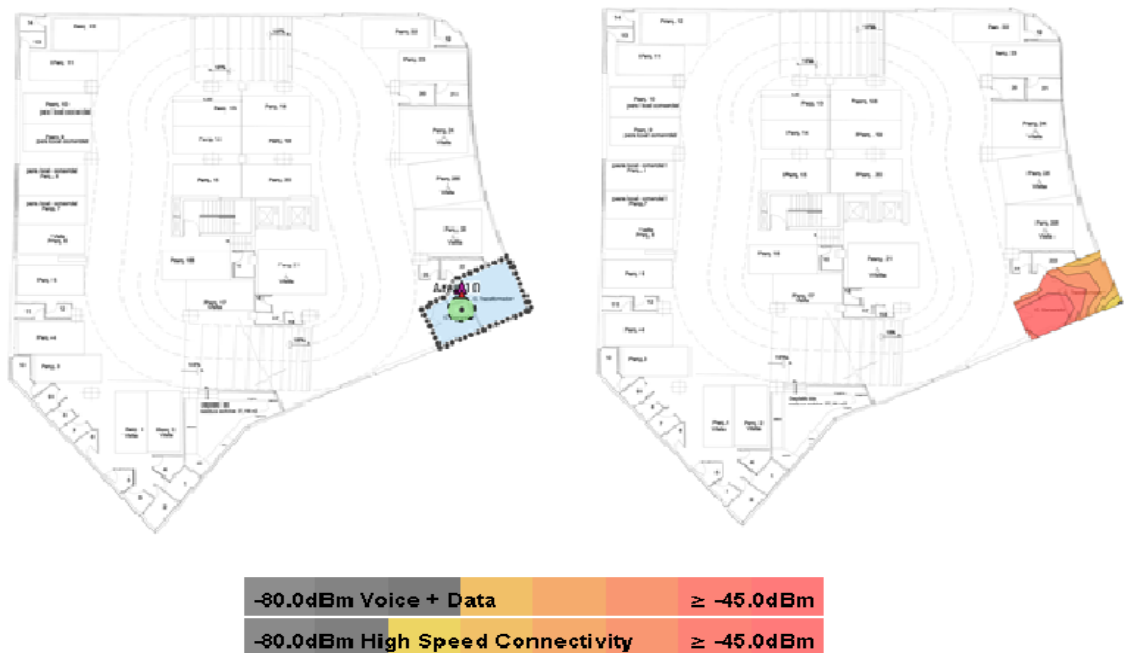
Ubicación de los puntos de acceso en el Subsuelo 4, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge

Mapa de intensidad de señal para el Subsuelo 3.



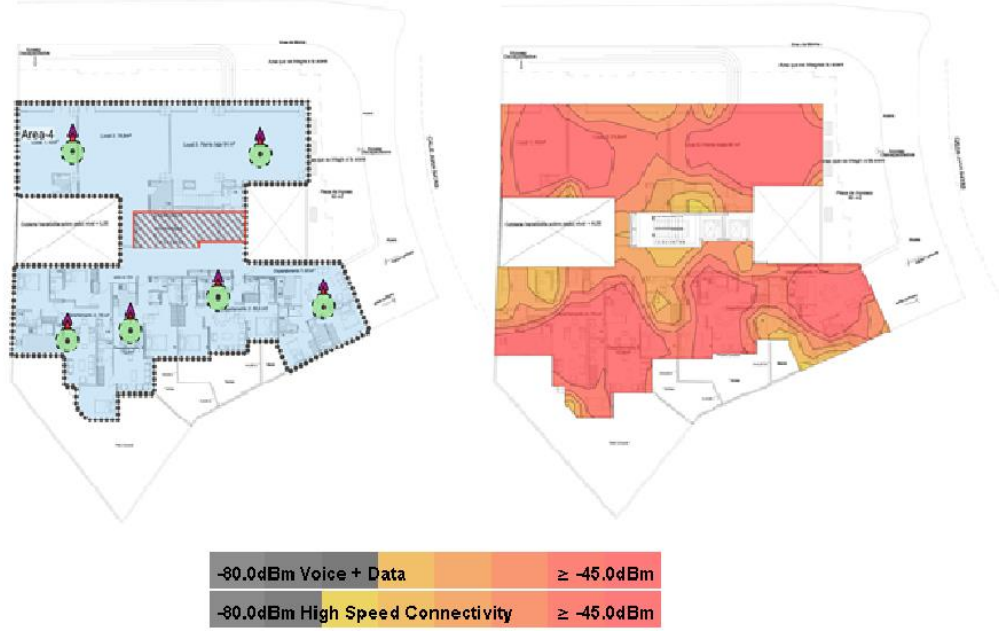
Ubicación de los puntos de acceso en el Subsuelo 3, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.

Mapa de intensidad de señal para el Subsuelo 1.



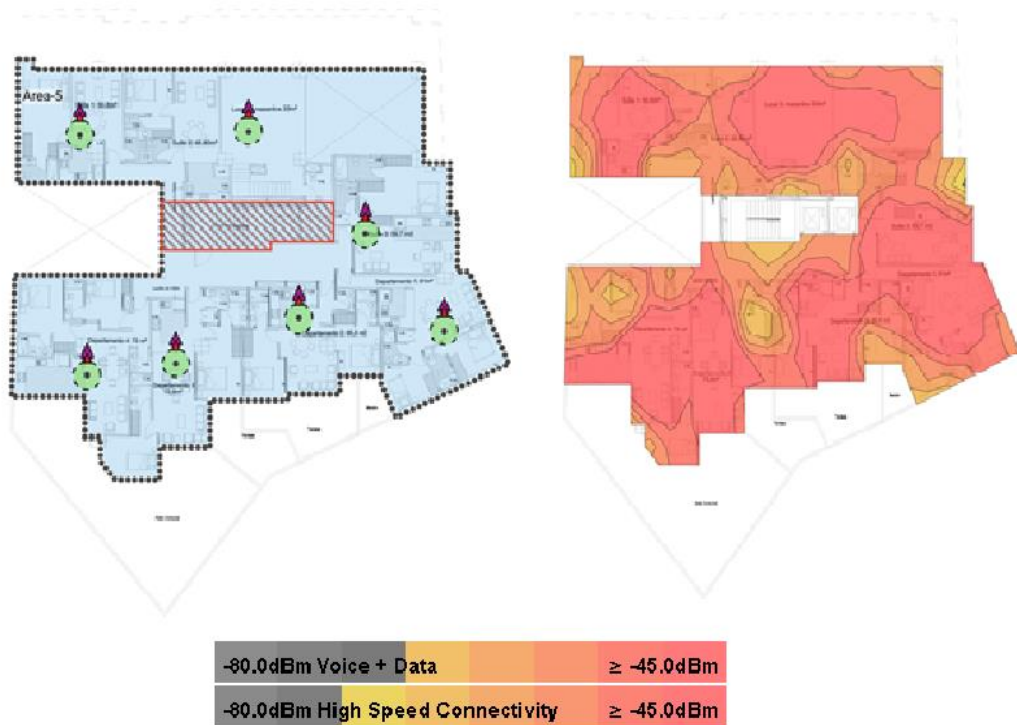
Ubicación de los puntos de acceso en el Subsuelo 1, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.

Mapa de intensidad de señal del Piso 1



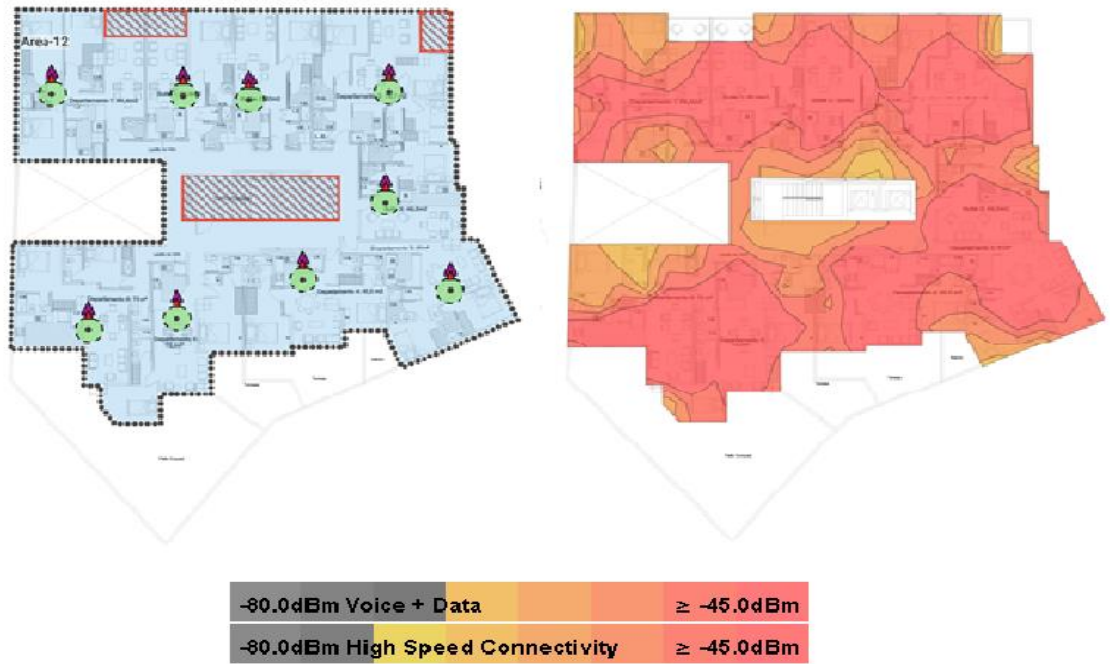
Ubicación de los puntos de acceso en el Piso 1, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.

Mapa de intensidad de señal para el Piso 2.



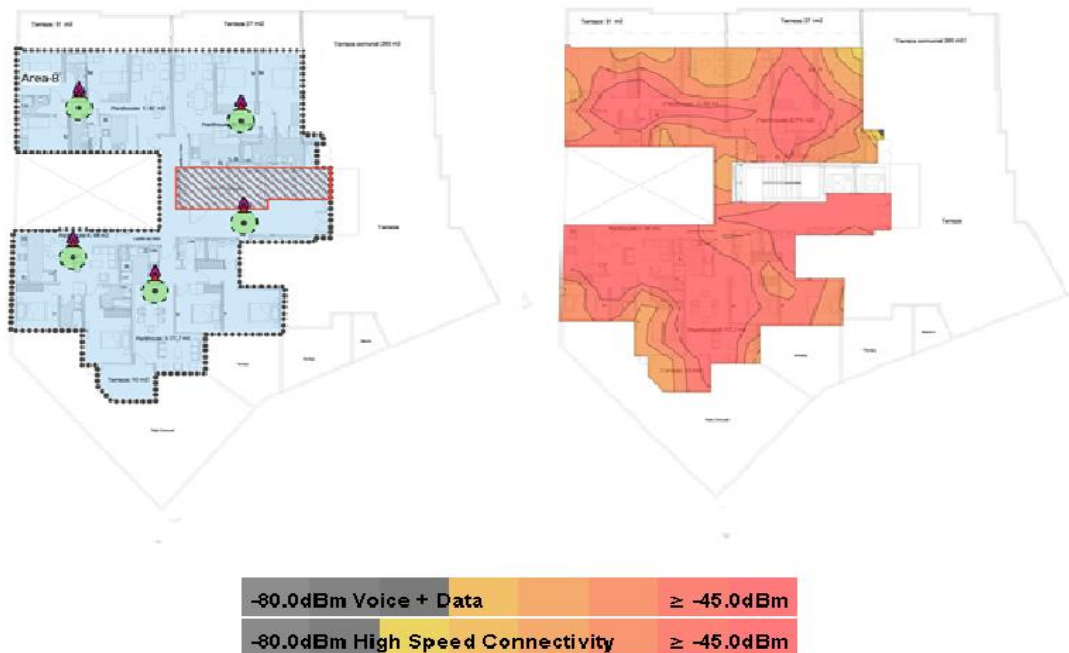
Ubicación de los puntos de acceso en el Piso 2, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.

Mapa de intensidad de señal del Piso 3 al Piso 11.



Ubicación de los puntos de acceso del Piso 3 al Piso 11, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.

Mapa de intensidad de señal para el Piso 12



Ubicación de los puntos de acceso en el Piso 12, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.



Mapa de intensidad de señal para la terraza.



Ubicación de los puntos de acceso en la Terraza, Fuente: Ekahau. Elaborado por: Chinachi Andrés y Rios Jorge.

**Anexo 9:** Cuadrante de Gartner.

Cuadrante de Gartner

Figure 1. Magic Quadrant for the Wired and Wireless LAN Access Infrastructure



Cuadrante de Gartner para infraestructuras de acceso LAN cableado e inalámbrico, Autor: (Bill Menezes, 2019)



## Anexo 10: Selección y comparación del equipamiento para la WLAN

### Selección de AP

Parámetros \ APs	Cisco Aironet 702i	Huawei AP4050DE-M	Extreme Networks AP305C	Aruba 303H
Menor precio	x			
Soporta estándares IEEE 802.11n/ac/a/b	x	x	x	x
Banda dual	x	x	x	x
Mayor ganancia de antenas y potencia de transmisión	x	x		x
Selección				

Tabla selección y comparación de puntos de acceso. Elaborado por: Chinachi Andrés y Rios Jorge

### Características de AP

Características Principales	Cisco	Huawei	Extreme Networks	Aruba
Modelo	Aironet 702i	AP4050DE-M	AP305C	ARUBA 303H
Estándares	IEEE 802.11n/ac/a/b, entre otros	IEEE 802.11a/b/g/h/k/ac, entre otros	IEEE 802.11a/n/g/b/ac/ax	IEEE 802.11a/b/g/n/ac/
Antena	Internas, 1 para 2,4GHz 2 para 5GHz	Interna Omnidireccional de banda dual	Interna Omnidireccional de banda dual	Dos integradas, banda dual direccionales
Ganancia de la antena	2.4 GHz: 3 dBi 5 GHz: 5 dBi	2.4G: 3 dBi 5G: 3 dBi	2.4 GHz: 3.75 dBi 5 GHz: 3.75 dBi	2.4 GHz: 4.3 dBi 5 GHz: 6.3 dBi
Frecuencias	2.4, 5 GHz	2.4, 5 GHz	2.4, 5 GHz	2.4, 5 GHz
Potencia de transmisión	2.4GHz: 5, 20 dBm. 5GHz: 5, 20 dBm	2.4G: 27 dBm 5G: 27 dBm	2.4GHz: 11, 18 dBm. 5GHz: 13, 18 dBm	2.4GHz: 18, 21 dBm. 5GHz: 16, 19 dBm
Precio	\$99.99	\$435	\$543	\$547

Tabla de características de puntos de acceso. Elaborado por: Chinachi Andrés y Rios Jorge.

### Selección de WLC (Wireless LAN Controller)

Parámetros \ WLCs	Cisco 3504	C9800-LC-K9	C9800-40-K9	Catalyst 9800-80-K9
Soporta estándares 802.11 a/b/g/d/h/n, wif 6	x	x	x	x
Menor precio	x			
Soporta el número de APs a configurar en el edificio (150)	x	x	x	x
Soporta APs Aironets	x	x	x	x
Selección				

Tabla de selección y comparación de WLC. Elaborado por: Chinachi Andrés y Rios Jorge.

Características de los WLCs

Características	Cisco			
	Cisco 3504	C9800-LC-K9	C9800-40-K9	Catalyst 9800-80-K9
<b>Estándares</b>	IEEE 802.11a/b/g/d/e/h/n /k/r/u/w/ac/ wave 1 y 2, wifi 6.	IEEE 802.11a/b/g/h/k/n/ac , entre otros	IEEE 802.11a/b/g/h/k/n /ac, entre otros	IEEE 802.11a/b/g/h /k/n/ac, entre otros
<b>APs máximos</b>	150	250	2000	6000
<b>Número máximo de clientes</b>	3000	5000	32	64
<b>WLAN máximas</b>	4096	4096	4096	4096
<b>Modos de implementación</b>	Centralizado, cisco flexconnect.	Centralizado, Cisco FlexConnect ® y fabric	Centralizado, Cisco FlexConnect ® y Fabric Wireless (SD-Access)	Centralized, Cisco FlexConnect ®, and Fabric Wireless (SD-Access)
<b>APs</b>	Aironet, catalyst.	Aironet 802.11ac Wave 1 y Wave 2, Catalyst 9100 802.11ax	Aironet 802.11ac Wave 1 y Wave 2, Catalyst 9100 802.11ax	Aironet, Wave 1 y Wave 2, Catalyst 9100 802.11ax
<b>Unidad de Rack</b>	1 U, con soporte	1 U	1 U	2 U
<b>Precio</b>	\$1590.00	\$ 2,203.10	\$ 5,599.00	\$ 7,000.00

Tabla de características de WLCs. Elaborado por: Chinachi Andrés y Rios Jorge.

**Anexo 11:** Protocolos de comunicación inalámbrica.

Comparación de Protocolos de Comunicación Inalámbrica

Parámetros	Protocolos de comunicación					
	WiFi	Zigbee	Bluetooth	LoRaWan	Sigfox	Z-Wave
<b>Redes de alcance corto y medio</b>	x		x			
<b>Bajo Costo</b>		x				
<b>Bajo consumo de energía</b>		x	x	x	x	x
<b>Compatibilidad con redes WSN</b>	x	x	x	x	x	x
<b>Altas velocidades</b>	x					
<b>Banda de frecuencia de 2,4GHz</b>	x	x	x			
<b>Gran ancho de Banda</b>	x					x
<b>Gran compatibilidad con dispositivos IoT del mercado</b>	x	x	x			x
<b>Selección</b>						

Tabla de selección y comparación de protocolos de comunicación inalámbrica. Elaborado por:

Chinachi Andrés, Rios Jorge.

Características de Protocolos de comunicación inalámbrica

Características	Wifi	Zigbee	Bluetooth	LoRaWan	Sigfox	Z-Wave
<b>Dirigidas</b>	Redes empresariales, residenciales, entornos públicos.	Redes de bajo costo, bajo consumo.	Redes de corto alcance, doméstica	Redes de largo alcance y bajo consumo	Redes de comunicación móvil, bajo consumo, baja velocidad y largo alcance	Redes domésticas de bajo consumo
<b>Norma</b>	802.11	802.15.4	802.15.1	LORA	Sigfox	Z-Wave
<b>Preferencia para WSN</b>	Si	Si	Si	Si	Si	Si
<b>Para redes</b>	WLAN	WPAN	WPAN	LPWAN	LPWAN	WPAN
<b>Alcance</b>	1-100 m	1-100 m	1-10 m	15 Km	20 Km	100 m
<b>Velocidad</b>	54 Mbps	250 Kbps	1 Mbps	0.3 a 5 kbps	12 bytes up, 8 bytes down	9.6, 40, 100 Kbps
<b>Banda de Frecuencia</b>	2,4 GHz	2,4 GHz	2,4 GHz	915, 868, 433 MHz	868, 902 MHz	900 MHz
<b>Ancho de Banda</b>	54000 Kbps	20-250 Kbps	720 Kbps	125 Kbps	100 Kbps	5 Mbps

Tabla de características de protocolos de comunicación inalámbrica. Elaborado por: Chinachi Andrés, Rios Jorge.

**Anexo 12:** Selección y comparación de equipamiento WSN

Selección de Nodos

Parámetros	Plataforma	Raspberry Pi	Arduino	NodeMCU
<b>Menor Costo</b>				x
<b>Diversas fuentes de alimentación</b>		x	x	x
<b>Trabaja con el estandar WiFi 802.11 n</b>		x		x
<b>Selección</b>				

Tabla de selección y comparación de nodos controladores. Elaborado por: Chinachi Andrés y Rios Jorge.

Características de Nodos

Características	Raspberry Pi	Arduino	NodeMCU
<b>Fabricante</b>	Raspberry Pi Foundation	Arduino	Amico
<b>Microprocesador</b>	Broadcom BCM2835 (procesador)	ATmega328P	Tensilica L106
<b>Alimentación</b>	5.1 V DC USB/ Fuente	USB / fuente 5V DC	USB 5V DC
<b>Memoria RAM</b>	512MB	2 KB	96 KB
<b>Memoria FLASH</b>	No incorporada	32 KB	4 MB
<b>WLAN</b>	802.11n	N.A.	802.11 b/g/n
<b>Tamaño</b>	65x30mm	80.01 x 55.11 x 24.89 mm	49 x 24.5 x 13 mm
<b>Modelo</b>	Zero W	UNO R3	V3
<b>Precio</b>	47.99 \$	9,90 \$	10 \$

Tabla de características de nodos controladores. Elaborado por: Chinachi Andrés y Rios Jorge.

### Selección de Sensores de gas butano y humo

<b>Sensor Butano y Humo</b>	<b>MQ-2</b>	<b>MQ-3</b>	<b>MQ-5</b>	<b>MQ-6</b>	<b>MQ-303A</b>	<b>MQ-306A</b>
<b>Parámetros</b>						
<b>Compatibilidad con la plataforma NodeMCU</b>	x	x	x	x	x	x
<b>Mayor rango de detección</b>	x	x	x	x		
<b>Detección de Gas GLP</b>	x		x	x		x
<b>Detección de Humo</b>	x	x			x	
<b>Selección</b>						

Tabla de selección y comparación de sensores de gas butano y humo. Elaborado por: Chinachi Andrés y Rios Jorge.

### Características de sensores de gas butano y humo

<b>Características</b>	<b>MQ-2</b>	<b>MQ-3</b>	<b>MQ-5</b>	<b>MQ-6</b>	<b>MQ-303A</b>	<b>MQ-306A</b>
<b>Voltaje (DC)</b>	5V	5V	5V	5V±0.1 V	0.9±0.1 V	0.9V
<b>Resistencia</b>	3-30 KΩ	1-8 MΩ	10-60 KΩ	10-60 KΩ	20-200 KΩ	20-200 KΩ
<b>Rango de detección</b>	300 – 10000 ppm	10 -1000 ppm	200 – 10000 ppm	200- 10000ppm	50-300 ppm	1000-3000 ppm
<b>Temperatura de operación</b>	-20 °C ~ +55 °C	-10 °C ~ +70 °C	-20 °C ~ +70 °C	20°C ~ +2 °C	20°C ~ ±2 °C	20°C ~ ±2 °C
<b>Consumo</b>	800 mW	700 mW	800 mW	≤ 750mW	≤ 130mW	≤ 120mW
<b>Sustancias detectadas</b>	Metano, butano, GLP, alcohol, humo	Alcohol, Etanol, humo	Gas natural, GLP, isobutano, propano	Isobutano, GLP, propano, GNL	Alcohol, etanol, humo	Butano, GLP, GNL, isobutano, H2, propano.
<b>Elección</b>						

Tabla de características de sensores de gas butano y humo. Elaborado por: Chinachi Andrés y Rios Jorge.

### Selección de Sensores de humedad y temperatura

<b>Sensor de temperatura</b>	<b>DHT11</b>	<b>DHT22</b>	<b>LM35DZ</b>	<b>TMP36</b>	<b>LM75</b>	<b>DS18B20</b>
<b>Parámetros</b>						
<b>Mayor rango de detección de temperatura</b>				x	x	x
<b>Incluye medición de humedad</b>	x	x				
<b>Precisión de humedad</b>	x	x				
<b>Mayor rango de detección de humedad</b>		x	x			
<b>Menor error de medición</b>		x	x	x		x
<b>Selección</b>						

Tabla de selección y comparación de sensores de humedad y temperatura. Elaborado por: Chinachi Andrés y Rios Jorge.

Características de sensores de humedad y temperatura

Características	DHT11	DHT22	LM35DZ	TMP36	LM75	DS18B20
Voltaje (DC)	5V	5V	4 – 30V	2,7-5,5 V	3-5V	3-5,5v
Rango de medición de temperatura	0 °C ~ 50 °C	-40 °C ~ 80 °C	-55 °C ~ 150 °C	40 °C ~ 125°C	-25°~100°C /-55° ~125°C: ±3°C	-55°C ~ 125°C
Rango de medición de humedad (RH)	20 – 90% RH	0 – 100% RH	-	-	-	-
Precisión de humedad	±5%RH	±5%RH	-	-	-	-
Precisión de temperatura	±2°C	±0.5°C	±0.5°C	±0.5°C	±2°C/±3°C	±0.5°C

Tabla de características de sensores de humedad y temperatura. Elaborado por: Chinachi Andrés y Rios Jorge.

Selección de Sensor de Fuego

Características	KY-026
Voltaje	3.3 – 5.3V DC
Espectro	760 – 1100 nm
Ángulo de detección	0 – 60°
Temperatura	-25 °C ~ 85 °C
Selección	

Tabla de características del sensor de fuego. Elaborado por: Chinachi Andrés y Rios Jorge.

Selección de Sensores de nivel

Sensor de Nivel / Parámetros	HC-SR04	JSN-SR04T	Water Sensor	Float Sensor Water
Posee ángulo de detección	x	x		
Mayor rango de distancia	x			
Selección				

Tabla de selección y comparación de sensores de nivel. Elaborado por: Chinachi Andrés y Rios Jorge.

Características	HC-SR04	JSN-SR04T	Water Sensor	Float sensor water level ZP4510
Voltaje	5V DC	5 V DC	5 V DC	220 V DC
Frecuencia de trabajo	40 KHz	40 KHz	-	
Corriente de trabajo	20 mA	30 mA	20 mA	500 mA
Rango de distancia	2 - 4 m	25-450 cm	Área de 40x16mm	-
Ángulo efectivo	< 15°	-	-	-
Ángulo de medición	30°	75°	-	-

Tabla de características de sensores de nivel. Elaborado por: Chinachi Andrés y Rios Jorge.

### Anexo 13: Selección y comparación de Protocolos de Comunicación IoT

Protocolos de Comunicación IoT

<b>Protocolo de mensajería</b>	<b>MQTT</b>	<b>AMQP</b>	<b>CoAP</b>	<b>HTTP</b>
<b>Parámetros</b>				
<b>Soporta QoS</b>	x	x		
<b>Utiliza Bajo ancho de Banda</b>	x		x	
<b>Diseñado para IoT</b>	x		x	
<b>Confiable y seguro</b>		x		x
<b>Alta disponibilidad</b>	x	x		x
<b>Menor latencia y consumo de recursos</b>	x	x		
<b>Modelo cliente-servidor</b>	x	x		
<b>Gran compatibilidad con dispositivos</b>	x	x		x
<b>Selección</b>				

Tabla de selección y comparación de protocolos de comunicación IoT. Elaborado por: Chinachi Andrés y Rios Jorge.

### Anexo 14: Selección y comparación de Base de Datos

Selección y comparación de Bases de Datos

<b>Base de datos</b>	<b>MariaDB</b>	<b>MySQL</b>	<b>PostgreSQL</b>	<b>InfluxDB</b>	<b>MongDB</b>
<b>Parámetros</b>					
<b>Disponibilidad y flexibilidad</b>	x	x	x	x	x
<b>Gran compatibilidad</b>	x	x	x	x	x
<b>Escalabe</b>	x	x		x	x
<b>Soporte</b>	x	x	x	x	x
<b>Tolerante a fallas</b>	x	x	x	x	x
<b>Datos en series de tiempo</b>		x		x	
<b>Diseñada para manejo de datos IoT y análisis en tiempo real</b>				x	
<b>Permiso de usuarios</b>	x	x	x	x	x
<b>Posibilidad de trabajar en memoria</b>				x	
<b>Trabaja con otras herramientas para la graficación y visualización de datos</b>				x	
<b>Manejo de datos no estructurados</b>				x	x
<b>Consultas rápidas</b>				x	x
<b>Maneja gran cantidad de datos</b>	x	x	x	x	x
<b>Selección</b>					

Tabla de selección y comparación de Bases de Datos existentes. Elaborado por: Chinachi Andrés y Rios Jorge.

## Anexo 15: Selección y comparación de Servidores

### Elección de Servidores

Parámetros	Servidores		
	HPE	DELL	LENOVO
Menor costo		x	
Posee memoria de almacenamiento		x	
Gran compatibilidad con sistemas operativos	x	x	
Mayor memoria RAM	x		x
Selección			

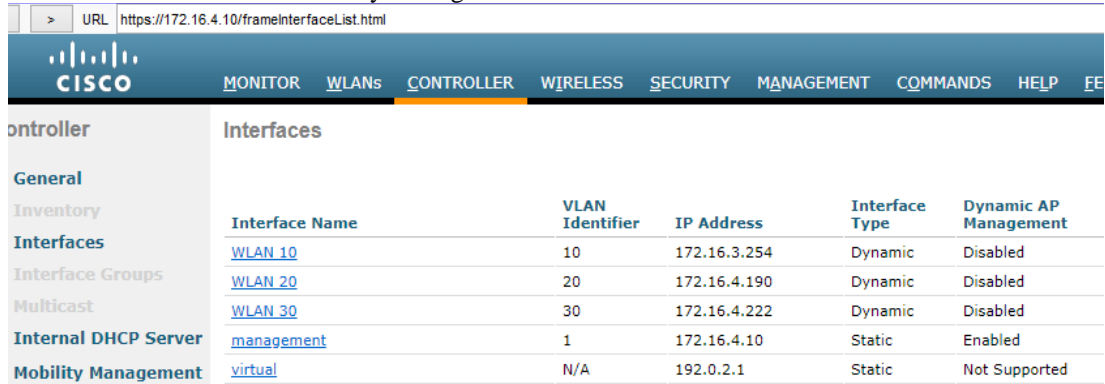
Tabla de comparación y selección de servidores. Elaborado por: Chinachi Andrés y Rios Jorge.

Características Principales	HPE	DELL	LENOVO
<b>Modelo</b>	HPE ProLiant DL360	PowerEdge R240	ThinkSystem SR530
<b>Procesador</b>	Intel® Xeon® Scalable 4110	Intel® Celeron® G4930	Intel Xeon Bronze 3106
<b>Núcleos</b>	8 núcleos, 2,1 GHz, 85 W	2 núcleos, 3,20 GHz, 54 W	8-core, 1.7 GHz, 85 W
<b>Sistema Operativo Compatible</b>	Ubuntu® Server LTS Citrix® Hypervisor Microsoft® Windows Server® LTSC with Hyper-V (2016 and 2019) Red Hat® Enterprise Linux SUSE® Linux Enterprise Server VMware® ESXi®	Canonical® Ubuntu® Server LTS Citrix® Hypervisor Microsoft® Windows Server® LTSC with Hyper-V (2016 and 2019) Red Hat® Enterprise Linux SUSE® Linux Enterprise Server VMware® ESXi®	Microsoft Windows Server, SUSE, Red Hat, VMware vSphere
<b>Memoria RAM</b>	RDIMM de 32 GB (2 x 16 GB)	DDR4 ECC UDIMM de 8 GB	DDR4 ECC DIMM 16 GB
<b>Memoria de Almacenamiento</b>	No incluido, admite 8 unidades SFF	1 TB, 4 x 3.5 hot-plug SAS/ SATA/SSD	No incluido, 8 unidades SAS/SATA Simple-Swap de 3,5"
<b>Unidad de Rack</b>	1 U	1 U	1 U
<b>Precio</b>	\$2000	\$800	\$1800

Tabla de comparación y elección de servidores. Elaborado por: Chinachi Andrés y Rios Jorge.

## Anexo 16: Configuraciones de la WLC.

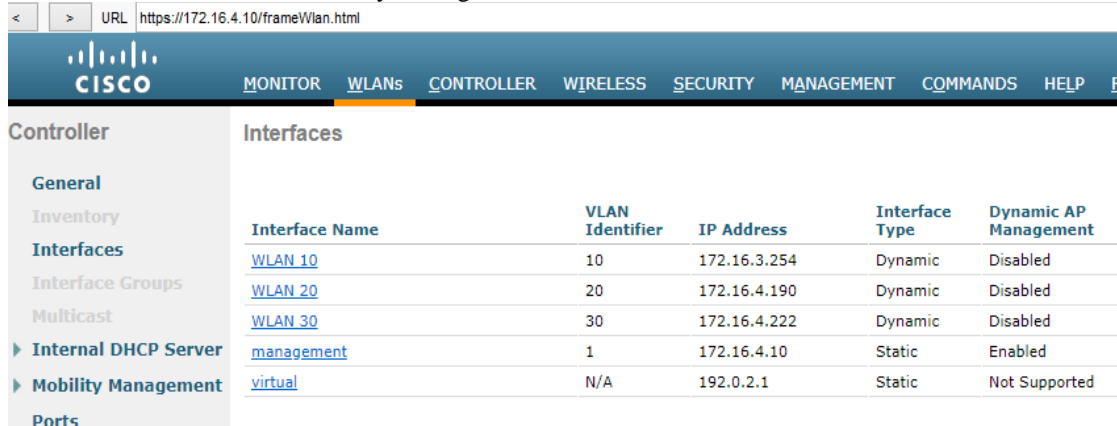
### Creación y Configuración de las interfaces en la WLC



Interface Name	VLAN Identifier	IP Address	Interface Type	Dynamic AP Management
<a href="#">WLAN 10</a>	10	172.16.3.254	Dynamic	Disabled
<a href="#">WLAN 20</a>	20	172.16.4.190	Dynamic	Disabled
<a href="#">WLAN 30</a>	30	172.16.4.222	Dynamic	Disabled
<a href="#">management</a>	1	172.16.4.10	Static	Enabled
<a href="#">virtual</a>	N/A	192.0.2.1	Static	Not Supported

Creación y configuración de las interfaces en la controladora WLC. Elaborado por: Chinachi Andrés y Rios Jorge.

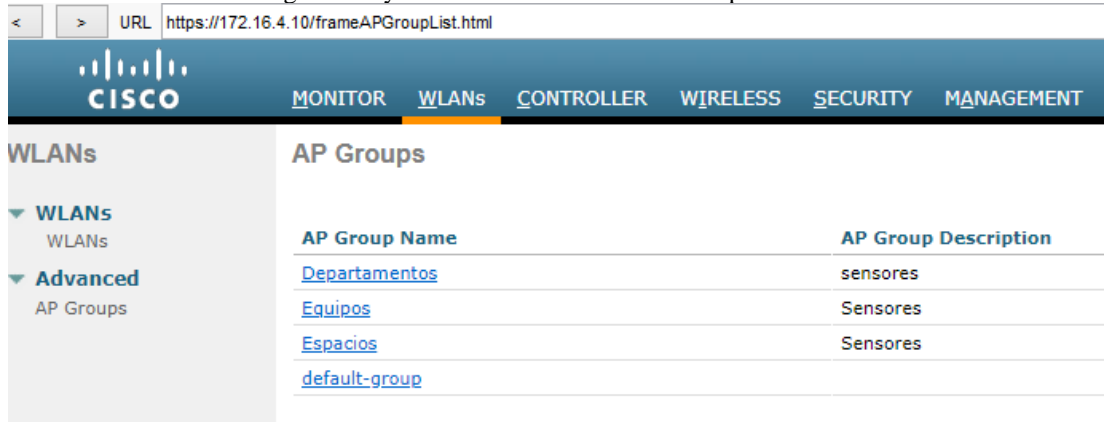
### Creación y Configuración de las WLANs en la WLC



Interface Name	VLAN Identifier	IP Address	Interface Type	Dynamic AP Management
<a href="#">WLAN 10</a>	10	172.16.3.254	Dynamic	Disabled
<a href="#">WLAN 20</a>	20	172.16.4.190	Dynamic	Disabled
<a href="#">WLAN 30</a>	30	172.16.4.222	Dynamic	Disabled
<a href="#">management</a>	1	172.16.4.10	Static	Enabled
<a href="#">virtual</a>	N/A	192.0.2.1	Static	Not Supported

Creación y configuración de las WLANs en la controladora WLC. Elaborado por: Chinachi Andrés y Rios Jorge.

### Configuración y vinculación de los APs en Grupos en la WLC



AP Group Name	AP Group Description
<a href="#">Departamentos</a>	sensores
<a href="#">Equipos</a>	Sensores
<a href="#">Espacios</a>	Sensores
<a href="#">default-group</a>	

Configuración y vinculación de grupos de APs en la controladora WLC. Elaborado por: Chinachi Andrés y Rios Jorge.



## Anexo 17: Programación de Sensores

### Programación sensor de temperatura

```
1. #include <ESP8266WiFi.h>
2. #include <PubSubClient.h>
3. #include "DHT.h"
4. #define wifi_ssid "XXXXXXX"
5. #define wifi_password "XXXXXXX"
6. #define mqtt_server "192.168.0.18"
7. #define mqtt_user "administrador"
8. #define mqtt_password "XXXXXX"
9. #define
   temperature_topic "Edificio/Planta_Baja/Gimnasio/sen
   nsor/temperatura"
10. #define
   humidity_topic "Edificio/Planta_Baja/Gimnasio/sens
   or/humidity"
11. #define
   switch_topic "Edificio/Planta_Baja/Gimnasio/ventila
   ción"
12. #define myClientID "T1"
13. #define myUsername "Sensor_temp_gimnasio"
14. #define myPassword "12345"
15. char message_buff[100];
16. long lastMsg = 0;
17. long lastRecu = 0;
18. bool debug = false;
19. int ventilacion_PIN = D7;
20. #define DHTPIN D2
21. #define DHTTYPE DHT22
22. DHT dht(DHTPIN, DHTTYPE);
23. WiFiClient espClient;
24. PubSubClient client(espClient);
25. void setup() {
26.   Serial.begin(9600);
27.   pinMode(ventilacion_PIN, OUTPUT);
28.   setup_wifi();
29.   client.setServer(mqtt_server, 1883);
30.   client.setCallback(callback);
31.   client.connect("myClientID", "myUsername", "myP
   assword");
32.   dht.begin();
33. }
34. void setup_wifi(){
35.   delay(10);
36.   Serial.println();
37.   Serial.print("Connecting to");
38.   Serial.println(wifi_ssid);
39.   WiFi.begin(wifi_ssid, wifi_password);
40.   while(WiFi.status() != WL_CONNECTED){
41.     delay(500);
42.     Serial.print(".");
43.   }
44.   Serial.println("");
45.   Serial.println("WiFi OK");
46.   Serial.println("=> ESP8266 IP address: ");
47.   Serial.print(WiFi.localIP());
48. }
49. void reconnect() {
50.   while(!client.connected()){
51.     client.connect("myClientID", "myUsername", "my
   Password");
52.     Serial.print("Connecting to MQTT broker ...");
53.     if(client.connect("ESP8266Client", mqtt_user,
   mqtt_password) && client.connect("myClientID", "
   myUsername", "myPassword")){
54.       Serial.println("OK");
55.     } else {
56.       Serial.print("OK, error: ");
57.       Serial.print(client.state());
58.       Serial.println("Wait 5 seconds before to retry");
59.       delay(5000);
60.     }
61.   }
62. }
63. void loop() {
64.   if(!client.connected()){
65.     reconnect();
66.   }
67.   client.loop();
68.   long now=millis();
69.   if(now - lastMsg > 60000){
70.     lastMsg = now;
71.     float h = dht.readHumidity();
72.     float t = dht.readTemperature();
73.     bool sent = false;
74.     if(isnan(t) || isnan(h)){
75.       Serial.println("KO, Please chek DHT sensor!");
76.       return;
77.     }
78.     Serial.println("Humedad ");
79.     Serial.print(h);
80.     Serial.print(" %t");
81.     Serial.println("Temperatura: ");
82.     Serial.print(t);
83.     Serial.print(" *C ");
84.     if(debug){
85.       Serial.print("Temperature: ");
86.       Serial.print(String(t).c_str());
87.       Serial.print(" | Humidity: ");
88.       Serial.print(String(h).c_str());
89.       sent = true;
90.     }
91.     client.publish(temperature_topic, String(t).c_str(), t
   rue);
92.     client.publish(humidity_topic, String
   (h).c_str(), true);
93.   }
94.   if(now - lastRecu > 1000){
95.     lastRecu = now;
96.     client.subscribe("Openhab/switch1");
97.   }
98. }
99. void callback(char* topic, byte* payload,
   unsigned int length){
100.   String topicStr = topic;
101.   if (topicStr == "Openhab/switch1")
102.   {
103.     if(payload[0] == '1'){
104.       digitalWrite(D7, HIGH);
105.       client.publish(switch_topic, "1", true);
106.     }
107.     else if (payload[0] == '0'){
108.       digitalWrite(D7, LOW);
109.       client.publish(switch_topic, "0", true);
110.     }
111.   }
112. }
```

Código sensor de temperatura y humedad. Elaborado por: Chinachi Andrés y Rios Jorge.

- Sensor de Humo

### Programación sensor de Humo

```

1. #include <ESP8266WiFi.h>
2. #include <PubSubClient.h>
3. #define wifi_ssid "XXXXXx"
4. #define wifi_password "XXXXXXXX"
5. #define mqtt_server "192.168.0.18"
6. #define mqtt_user "administrador"
7. #define mqtt_password "XXXXXX"
8. #define
   estado_topic "Edificio/Planta_Baja/Salón_Comunal/s
   ensor/humo"
9. #define
   estado_topic2 "Edificio/Planta_Baja/Salón_Comunal/64
   sensor/humo-ppm"
10. #define HUMO_PIN D0
11. #define myClientID "humo1"
12. #define myUsername "Sensor_humo_scomunal"
13. #define myPassword "12345"
14. char message_buff[100];
15. long lastMsg = 0;
16. long lastRecu = 0;
17. bool debug = false;
18. int digitalValue = -1;
19. int BUZZER_PIN = D7;
20. int LED_PIN = D2;
21. WiFiClient espClient;
22. PubSubClient client(espClient);
23. void setup() {
24.   Serial.begin(9600);
25.   pinMode(BUZZER_PIN,OUTPUT);
26.   pinMode(LED_PIN,OUTPUT);
27.   pinMode(HUMO_PIN, INPUT);
28.   setup_wifi();
29.   client.setServer(mqtt_server, 1883);
30.   client.connect("myClientID", "myUsername", "myP
   assword");
31.   Serial.println("Detección de humo con MQ-2...");
32. }
33. void setup_wifi(){
34.   delay(10);
35.   Serial.println();
36.   Serial.print("Connecting to");
37.   Serial.println(wifi_ssid);
38.   WiFi.begin(wifi_ssid, wifi_password);
39.   while(WiFi.status() != WL_CONNECTED){
40.     delay(500);
41.     Serial.print(".");
42.   }
43.   Serial.println("");
44.   Serial.println("WiFi OK");
45.   Serial.println("=> ESP8266 IP address: ");
46.   Serial.print(WiFi.localIP());
47. }
48. void reconnect() {
49.   while(!client.connected()){
50.     Serial.print("Connecting to MQTT broker ...");
51.     client.connect("myClientID", "myUsername", "my
   Password");
52.     if(client.connect("ESP8266Client", mqtt_user,
   mqtt_password)&& client.connect("myClientID", "m
   yUsername", "myPassword")){
53.       Serial.println("OK");
54.     }else{
55.       Serial.print("OK, error: ");
56.       Serial.print(client.state());
57.       Serial.println("Wait 5 seconds before to retry");
58.       delay(5000);
59.     }
60.   }
61. }
62. void loop() {
63.   if(!client.connected()){
64.     reconnect();
65.   }
66.   client.loop();
67.   long now=millis();
68.   if(now - lastMsg > 15000){
69.     lastMsg = now;
70.     int adc_MQ = analogRead(A0
71.     float voltaje = adc_MQ * (5.0 / 1023.0);
72.     float Rs=5000*((5-voltaje)/voltaje);
73.     double smoke=4280.3*pow(Rs/1746.63571, -
   2.394);
74.     client.publish(estado_topic2, String(smoke).c_str()
   , true);
75.     bool digitalNew = digitalRead(HUMO_PIN);
76.     bool sent = false;
77.     if (digitalNew != digitalValue) {
78.       digitalValue = digitalNew;
79.       Serial.println(String("Nuevo estado del módulo:
   ") + digitalValue);
80.     }
81.     if(isnan(digitalNew)){
82.       Serial.println("KO, Please chek MQ2 sensor!");
83.       return;
84.     }
85.     if (digitalValue){
86.       digitalWrite(BUZZER_PIN, HIGH);
87.       digitalWrite(LED_PIN, HIGH);
88.       client.publish(estado_topic, String("Sin presencia
   de HUMO").c_str(), true);
89.     }
90.   }else{
91.     digitalWrite(BUZZER_PIN, LOW);
92.     digitalWrite(LED_PIN, LOW);
93.     client.publish(estado_topic, String("Presencia de
   HUMO Detectado").c_str(), true);
94.   }
95.   /////Impresión de medidas del sensor
96.   Serial.print("adc:");
97.   Serial.print(adc_MQ);
98.   Serial.print(" voltaje:");
99.   Serial.print(voltaje);
100.  Serial.print(" Rs:");
101.  Serial.print(Rs);
102.  Serial.print(" smoke:");
103.  Serial.print(smoke);
104.  Serial.println("ppm");
105. }
106. }

```

Código sensor de humo. Elaborado por: Chinachi Andrés y Rios Jorge



- Sensor de Fuego

Programación sensor de Fuego

```

1. #include <ESP8266WiFi.h>
2. #include <PubSubClient.h>
3. #define wifi_ssid "XXXXXX"
4. #define wifi_password "XXXXXX"
5. #define mqtt_server "192.168.0.18"
6. #define mqtt_user "administrador"
7. #define mqtt_password "XXXXXX"
8. #define estado_topic "Edificio/Planta_Baja/Lavandería/sensor/llama"
9. #define LLAMA_PIN D0
10. #define myClientID "llama1"
11. #define myUsername "Sensor_llama_lavanderia"
12. #define myPassword "12345"
13. char message_buff[100];
14. long lastMsg = 0;
15. long lastRecu = 0;
16. bool debug = false;
17. int BUZZER_PIN = D7;
18. int LED_PIN = D5;
19. WiFiClient espClient;
20. PubSubClient client(espClient);
21. void setup() {
22.   Serial.begin(9600);
23.   pinMode(BUZZER_PIN,OUTPUT);
24.   pinMode(LED_PIN,OUTPUT);
25.   pinMode(LLAMA_PIN, INPUT);
26.   client.setServer(mqtt_server, 1883);
27.   client.connect("myClientID", "myUsername", "myPassword");
28.   Serial.println("Deteccion de Fuego con KY026...");
29. }
30. void setup_wifi(){
31.   delay(10);
32.   Serial.println();
33.   Serial.print("Connecting to");
34.   Serial.println(wifi_ssid);
35.   WiFi.begin(wifi_ssid, wifi_password);
36.   while(WiFi.status() != WL_CONNECTED){
37.     delay(500);
38.     Serial.print(".");
39.   }
40.   Serial.println("");
41.   Serial.println("WiFi OK");
42.   Serial.println("=> ESP8266 IP address: ");
43.   Serial.print(WiFi.localIP());
44. }
45. void reconnect() {
46.   while(!client.connected()){
47.     Serial.print("Connecting to MQTT broker ...");
48.     client.connect("myClientID", "myUsername", "myPassword");
49.     if(client.connect("ESP8266Client", mqtt_user, mqtt_password)&&client.connect("myClientID", "myUsername", "myPassword")){
50.       Serial.println("OK");
51.     }else{
52.       Serial.print("OK, error: ");
53.       Serial.print(client.state());
54.       Serial.println("Wait 5 seconds before to retry");
55.       delay(5000);
56.     }
57.   }
58. }
59. void loop() {
60.   if(!client.connected()){
61.     reconnect();
62.   }
63.   client.loop();
64.   long now=millis();
65.   if(now - lastMsg > 15000){
66.     lastMsg = now;
67.     int digitalNew = digitalRead(LLAMA_PIN);
68.     bool sent = false;
69.     if (isnan(digitalNew)){
70.       Serial.println("KO, Please chek KY026 sensor!");
71.       return;
72.     }
73.     if (digitalNew){
74.       digitalWrite(LED_PIN, HIGH);
75.       client.publish(estado_topic, String("Sin presencia de LLAMA").c_str(), true);
76.       Serial.println("No hay llama");
77.       digitalWrite(BUZZER_PIN, HIGH);
78.       digitalWrite(LED_PIN, LOW);
79.     }else{
80.       digitalWrite(LED_PIN, HIGH);
81.       Serial.println("Hay llama");
82.       client.publish(estado_topic, String("Presencia de LLAMA Detectado").c_str(), true);
83.       digitalWrite(BUZZER_PIN, LOW);
84.     }
85.   }
86.   true;}}

```

Código sensor de fuego. Elaborado por: Chinachi Andrés y Rios Jorge.

- Sensor de Gas

### Programación sensor de Gas

```

1. #include <ESP8266WiFi.h>
2. #include <PubSubClient.h>
3. #define wifi_ssid "XXXXXXX"
4. #define wifi_password "XXXXXXX"
5. #define mqtt_server "192.168.0.18"
6. #define mqtt_user "administrador"
7. #define mqtt_password "2679674"
8. #define
   estado_topic "Edificio/Planta_Baja/Conserje/sensor/
   GLP"
9. #define
   estado_topic2 "Edificio/Planta_Baja/Conserje/sensor/
   GLP_PPM"
10. #define GAS_PIN D0
11. #define myClientID "GLP1"
12. #define myUsername "Sensor_glp_conserje"
13. #define myPassword "12345"
14. char message_buff[100];
15. long lastMsg = 0;
16. long lastRecu = 0;
17. bool debug = false;
18. int digitalValue = -1;
19. int BUZZER_PIN = D7;
20. int LED_PIN = D5;
21. WiFiClient espClient;
22. PubSubClient client(espClient);
23. void setup() {
24.   Serial.begin(9600);
25.   pinMode(BUZZER_PIN, OUTPUT);
26.   pinMode(LED_PIN, OUTPUT);
27.   pinMode(GAS_PIN, INPUT);
28.   client.setServer(mqtt_server, 1883);
29.   client.connect("myClientID", "myUsername", "myP
   assword");
30.   Serial.println("Deteccion de gas con MQ-2...");
31. }
32. void setup_wifi(){
33.   delay(10);
34.   Serial.println();
35.   Serial.print("Connecting to");
36.   Serial.println(wifi_ssid);
37.   WiFi.begin(wifi_ssid, wifi_password);
38.   while(WiFi.status() != WL_CONNECTED){
39.     delay(500);
40.     Serial.print(".");
41.   }
42.   Serial.println("");
43.   Serial.println("WiFi OK");
44.   Serial.println("=> ESP8266 IP address: ");
45.   Serial.print(WiFi.localIP());
46. }
47. void reconnect() {
48.   while(!client.connected()){
49.     client.connect("myClientID", "myUsername", "my
   Password");
50.     Serial.print("Connecting to MQTT broker ...");
51.     if(client.connect("ESP8266Client", mqtt_user,
   mqtt_password)&& client.connect("myClientID", "m
   yUsername", "myPassword")){
52.       Serial.println("OK");
53.     }else{
54.       Serial.print("OK, error: ");
55.       Serial.print(client.state());
56.       Serial.println("Wait 5 seconds before to retry");
57.       delay(5000);
58.     }
59.   }
60. }
61. void loop() {
62.   if(!client.connected()){
63.     reconnect();
64.   }
65.   client.loop();
66.   long now=millis();
67.   if(now - lastMsg > 15000){
68.     lastMsg = now;
69.     int adc_MQ = analogRead(A0);
70.     float voltaje = adc_MQ * (5.0 / 1023.0);
71.     float Rs=2000*((5-voltaje)/voltaje);
72.     double GLP=1013*pow(Rs/130277.157, -2.343);
73.     client.publish(estado_topic2, String(GLP).c_str(), t
   rue); //Publicar valor de sensor
74.     bool digitalNew= digitalRead(GAS_PIN);
75.     bool sent = false;
76.     if (digitalNew != digitalValue) {
77.       digitalValue = digitalNew;
78.       Serial.println(String("Nuevo estado del modulo:
   ") + digitalValue);
79.     }
80.     if(isnan(digitalNew)){
81.       Serial.println("KO, Please chez MQ6 sensor!");
82.       return;
83.     }
84.     if (digitalNew){
85.       digitalWrite(BUZZER_PIN, HIGH);
86.       digitalWrite(LED_PIN, HIGH);
87.       client.publish(estado_topic, String("Sin presencia
   de GAS").c_str(), true);
88.     }
89.     }else{
90.       digitalWrite(BUZZER_PIN, LOW);
91.       digitalWrite(LED_PIN, LOW);
92.       client.publish(estado_topic, String("Presencia de
   GAS Detectado").c_str(), true);
93.     }
94.     Serial.print("adc:");
95.     Serial.print(adc_MQ);
96.     Serial.print(" voltaje:");
97.     Serial.print(voltaje);
98.     Serial.print(" Rs:");
99.     Serial.print(Rs);
100.    Serial.print(" GLP:");
101.    Serial.print(GLP);
102.    Serial.println("ppm");
103.  }
104. }

```

Código sensor de gas. Elaborado por: Chinachi Andrés y Rios Jorge.

# Anexo 18: Integración MQTT

## Configuración MQTT Broker

Configuration > Things > Edit > Broker

**openHAB**

Control

Inbox 1

Configuration

System

Bindings

Services

Things

Items

Add-ons

Preferences

Paper UI

Name

Broker

Location

Planta Baja

Configuration Parameters

Configure parameters for the thing.

Broker Hostname/IP

192.168.0.18

The IP/Hostname of the MQTT broker

Secure Connection

Uses TLS/SSL to establish a secure connection to the broker.

SHOW LESS

Broker Port

Quality of Service

At most once (0)

The port is optional, if none is provided, the typical ports 1883 and 8883 (SSL) are used.

Client ID

Reconnect Time

Configuration > Things > Edit > Broker

**openHAB**

Control

Inbox 1

Configuration

System

Bindings

Services

Things

Items

Add-ons

Preferences

Paper UI

Broker Port

1883

The port is optional, if none is provided, the typical ports 1883 and 8883 (SSL) are used.

Quality of Service

At most once (0)

Client ID

Use a fixed client ID. Defaults to empty which means a client ID is generated for this connection.

Reconnect Time

60000

Reconnect time in ms. If a connection is lost, the binding will wait this time before it tries to reconnect.

Heartbeat

60

Keep alive / heartbeat timer in s. It can take up to this time to determine if a server connection is lost. A lower value may keep the broker unnecessarily busy for no or little additional value.

Last Will Message

The last will message.

less

Last Will Topic

Defaults to empty and therefore disables the last will.

Last Will QoS

At most once (0)

The quality of service parameter of the last will.

Last Will Retain

True if last Will should be retained (defaults to false)

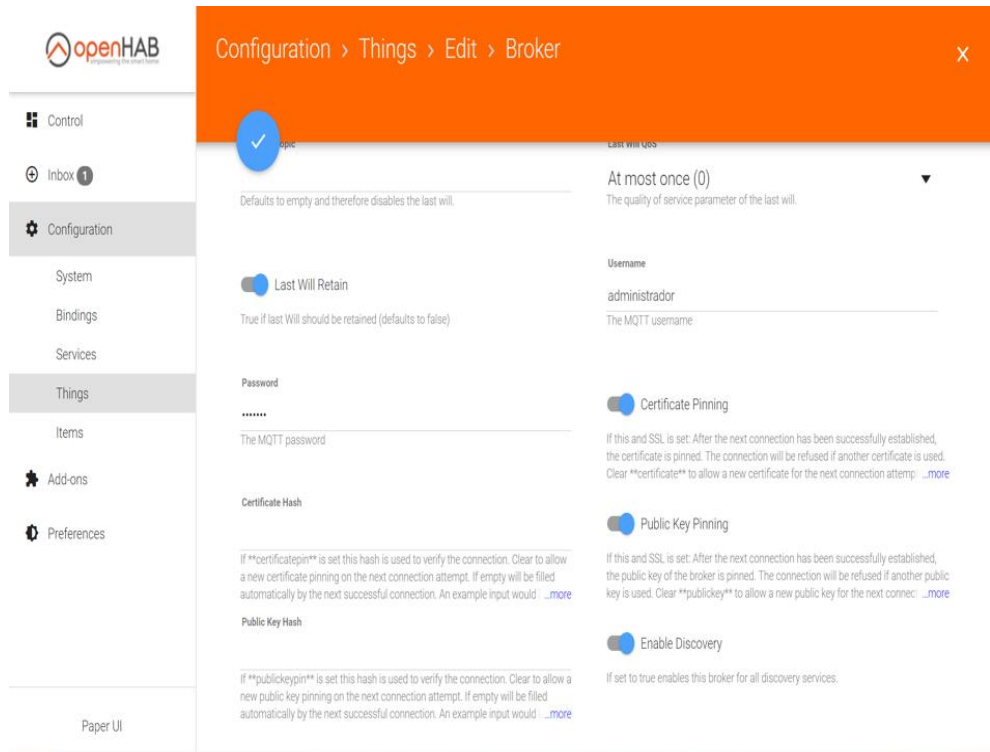
Username

administrador

The MQTT username

Password





Configuración MQTT en la plataforma OpenHab. Elaborado por: Chinachi Andrés y Rios Jorge.

## Anexo 19: Configuración rules y telegram.cfg

Archivo de configuración telegram.cfg

```
bots=Administrador_SCH
Administrador_SCH.chatId=1238766117
Administrador_SCH.token=1185297844:AAG1h6dRPFzhUvGFpAKpvwz-ZS7gDwaTR6k
```

Configuración del archivo telegram.cfg en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

### Archivo de configuración rules.cfg

```
rule "Notification Nivel de agua cisterna"
when
  Item CisternaNivelCisterna changed
then
  if (CisternaNivelCisterna.state <= 30 && previousState >30 )
  {
    logInfo("Sensor Nivel Cisterna Subsuelo", "Nivel Bajo")
    sendBroadcastNotification("Bajo nivel en tanque"+" "+ CisternaNivelCisterna.s
tate.format("%.1f") + " %")
    sendTelegram("Administrador_SCH", "BAJO NIVEL DE AGUA EN LA CISTERNA")
    sendHttpRequest("https://api.callmebot.com/whatsapp.php?source=OpenHAB&pho
ne="+593999797890&apikey=657312&text=Bajo+Nivel+en+la+Cisterna")
  }
end
//Sensor temperatura
rule "Notification Temperatura Gimnasio"
when
  Item GimnasioGenericMQTTThingTemperature changed
then
  if (GimnasioGenericMQTTThingTemperature.state >= 24 && previousState < 24 )
  {
    logInfo("Sensor Temperatura Gimnasio", "Alta temperatura")
    sendBroadcastNotification("Alta Temperatura"+" "+ GimnasioGenericMQTTThingTem
perature.state.format("%.1f") + " C")
    sendTelegram("Administrador_SCH", "ALTA TEMPERATURA EN EL GIMNASIO")
    sendHttpRequest("https://api.callmebot.com/whatsapp.php?source=OpenHAB&pho
ne="+593999797890&apikey=657312&text=Alta+TEMPERATURA+en+EL+GIMNASIO")
  }
end
//gas glp
rule "Notification Llama Lavanderia"
when
  Item LavanderiaSensorFirePB changed
then
  if (LavanderiaSensorFirePB.state == "Presencia de LLAMA Detectado" && previousSta
te == "Sin presencia de LLAMA")
  {
    logInfo("Sensor de Llama Lavanderia", "Presencia de Llama")
    sendBroadcastNotification("presencia de Llama")
    sendTelegram("Administrador_SCH", "LLAMA EN LA LAVANDERÍA")
    sendHttpRequest("https://api.callmebot.com/whatsapp.php?source=OpenHAB&pho
ne="+593999797890&apikey=657312&text=llama+en+la+lavanderia")
  }
}
```

Configuración del archivo rules.cfg en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

### Anexo 20: Archivo de configuración influxdb.cfg

#### Configuración influxdb.cfg

```
url=http://192.168.1.19:8086
user=openhab
password=openhab
db=openhab_db
retentionPolicy=autogen
```

Configuración del archivo influxdb.cfg en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

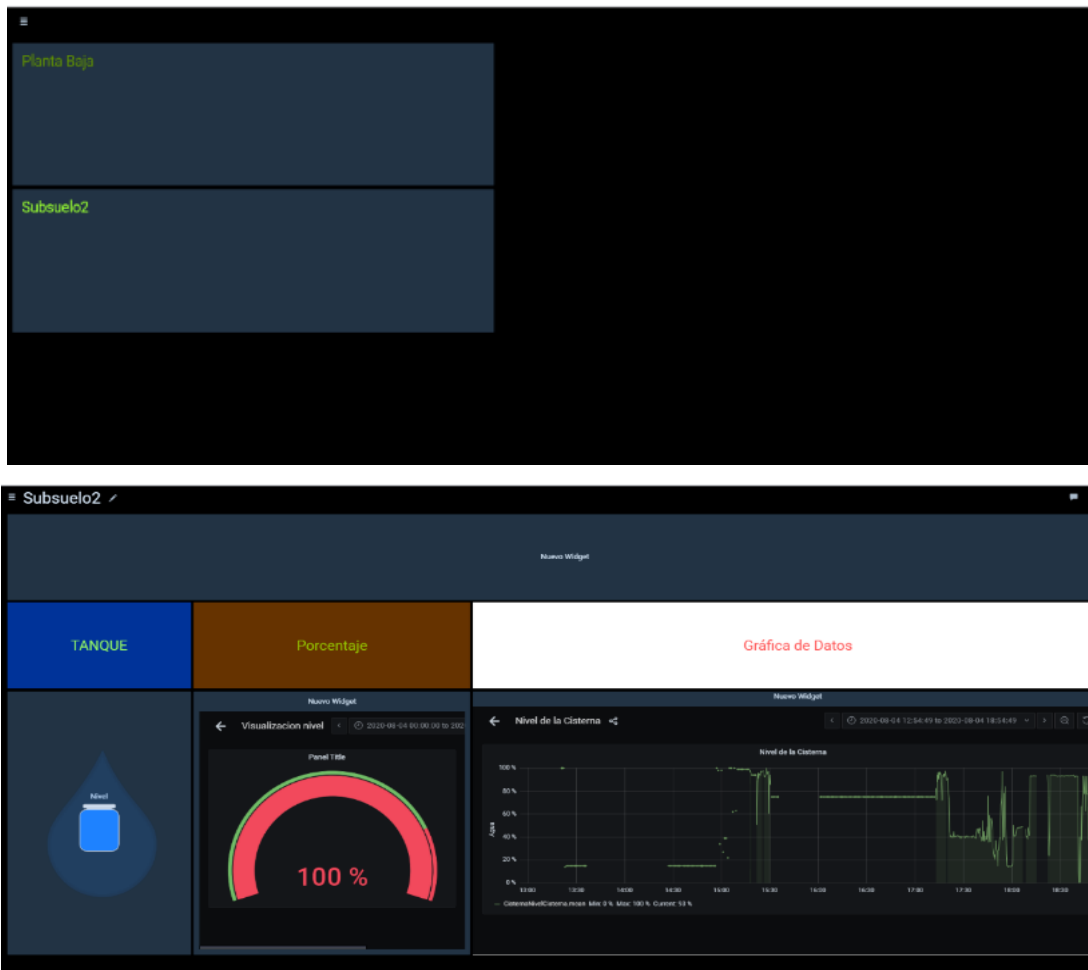
## Configuración influxdb.persist

```
Strategies
{
  everyMinute : "0 * * * * ?"
  everyHour : "0 0 * * * ?"
  everyDay : "0 0 0 * * ?"
  default = everyChange
}
Items
{
  CisternaNivelCisterna : strategy = everyChange, everyHour, everyDay, everyMinute
  SaloncomunalSensorHumoPB : strategy = everyChange, everyMinute
  SaloncomunalSensorHumoPBppm : strategy = everyChange, everyMinute
  GimnasioGenericMQTTThingTemperature : strategy = everyChange, everyMinute
  GimnasioGenericMQTTThingHumidity : strategy = everyChange, everyMinute
  ConserjeSensorGLPPB : strategy = everyChange, everyMinute
  ConserjeSensorGLPPBppm : strategy = everyChange, everyMinute
  LavanderiaSensorFirePB : strategy = everyChange, everyMinute
}
```

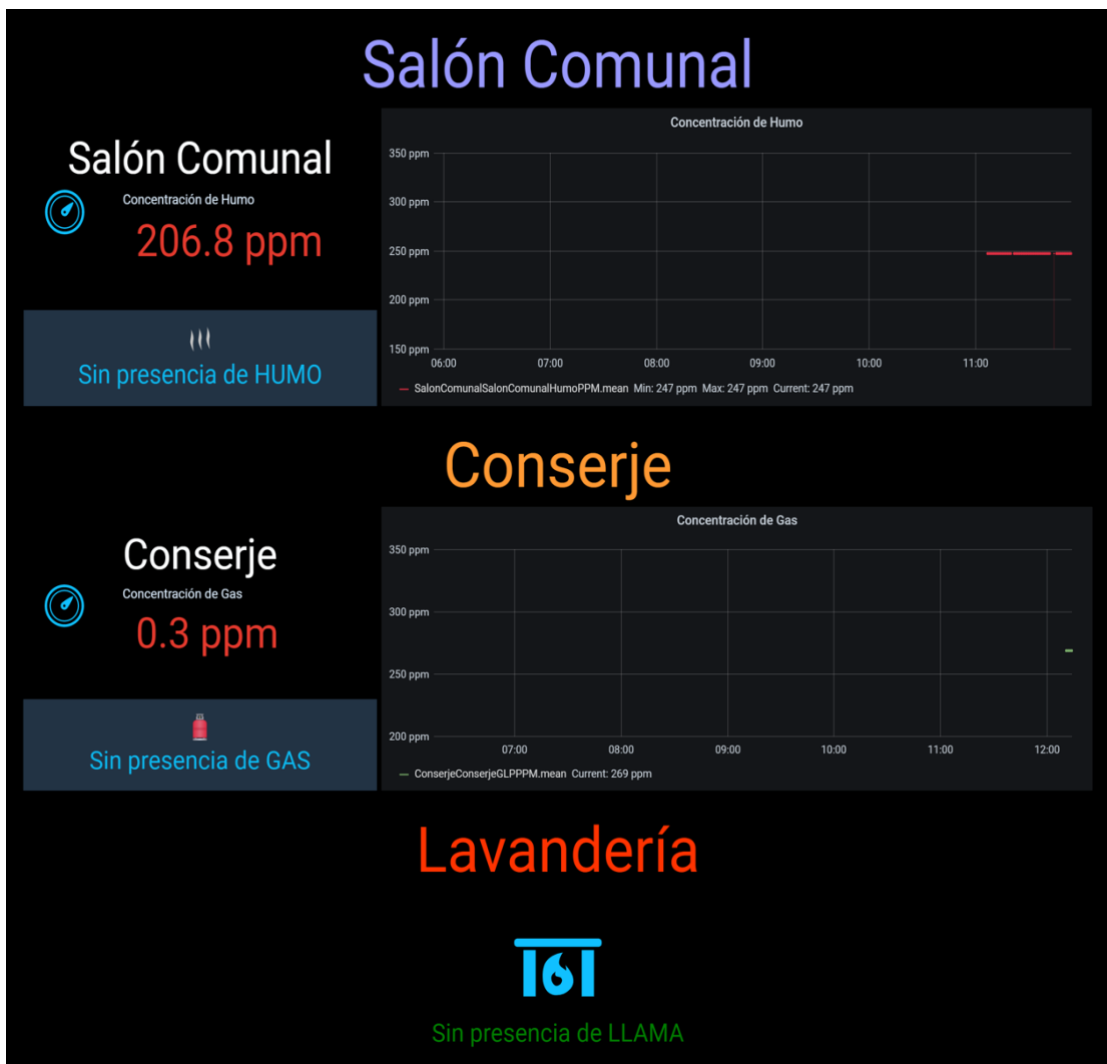
Configuración del archivo influxdb.persist en Visual Studio Code. Elaborado por: Chinachi Andrés y Rios Jorge.

## Anexo 21: Interfaz HabPanel

Paneles creados en HabPanel







Paneles creados en la interfaz gráfica HabPanel. Elaborado por: Chinachi Andrés y Rios Jorge.