



**Implementación de una red de sensores inalámbricos para la monitorización de variables físicas en cultivos hidropónicos mediante tecnología LoRa**

Loya Lasluisa Jhonatan Andrés

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica Y  
Telecomunicaciones

Ing. Lara Cueva, Román Alcides PhD

21 de enero del 2022



TESIS\_ESCRITO\_CORREGIDO5.pdf

Scanned on: 17:48 January 10, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text



Identical Words	515
Words with Minor Changes	90
Paraphrased Words	348
Ommited Words	0



**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES**

### **CERTIFICACIÓN**

Certifico que el trabajo de titulación, **“Implementación de una red de sensores inalámbricos para la monitorización de variables físicas en cultivos hidropónicos mediante tecnología LoRa”**, fue realizado por el señor **Loya Lasluisa, Jhonatan Andrés**; el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto, cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 21 de enero del 2022



Firmado electrónicamente por:

**ROMAN  
ALCIDES LARA**

**Dr. Lara Cueva, Román Alcides PhD.**

C.C 1713988218



**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES**  
**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES**

**RESPONSABILIDAD DE AUTORÍA**

Yo, **Loya Lasluisa Jhonatan Andrés**, con cédula de ciudadanía N° 1726074964, declaro que el contenido, ideas y criterios del trabajo de titulación: **Implementación de una red de sensores inalámbricos para la monitorización de variables físicas en cultivos hidropónicos mediante tecnología LoRa** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 21 de enero del 2022

.....  
**Loya Lasluisa, Jhonatan Andrés**

C.C 1726074964



**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES  
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES**

**AUTORIZACIÓN DE PUBLICACIÓN**

Yo, **Loya Lasluisa, Jhonatan Andrés**, con cédula de ciudadanía N° 1726074964, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Implementación de una red de sensores inalámbricos para la monitorización de variables físicas en cultivos hidropónicos mediante tecnología LoRa** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 21 de enero del 2022

.....  
**Loya Lasluisa, Jhonatan Andrés**

C. C 1726074964

## Dedicatoria

Dedico mi tesis con todo mi corazón a mis padres Luis y Yolanda que, con su amor incondicional, sacrificio y paciencia me han cuidado y guiado a lo largo de mi vida por el camino del bien, gracias por enseñarme a no temer a las adversidades porque Dios me acompaña siempre.

A Nicolle porque desde el momento en que te conocí mi vida ha sido extraordinaria, por enseñarme más sobre mí mismo, brindarme tu amor, creer en mí y apoyarme sin esperar nada a cambio. Eres la alegría de mi corazón.

A mi tío Juan Carlos por sus consejos éticos y humanos, prepararme para realizar los trabajos de la vida, por grandes o pequeños que sean, lo mejor posible, estoy tan orgulloso de ti. A mi abuelita Lucinda, tía Nelly y hermano Fabián porque con sus oraciones, amor y palabras de aliento me han motivado a ser una persona perseverante y que trabaja duro para cumplir sus metas.

A todos mis amigos pasados y presentes, en especial a Diego y Bryan por compartir conmigo experiencias buenas y malas, alegrías y tristezas a lo largo de estos años de amistad. A William, Richard y Alejandro por ser excelentes personas, compartir sus conocimientos, amistad y sobre todo muchas vivencias que hicieron de nuestro viaje universitario más ameno.

**Jhonatan Andrés**

## **Agradecimientos**

A Dios, Padre Misericordioso quien me ha cuidado, guiado y ha sido mi fortaleza en los momentos de mayor dificultad. A mis padres por proporcionarme todas las cosas que he necesitado y darme la oportunidad de tener una educación excelente en esta prestigiosa Universidad. Han trabajado duro por proporcionarme lo mejor y su apoyo fue fundamental para la culminación de este proyecto de tesis.

Mi sincero agradecimiento al director de este proyecto, Dr. Román Lara, por la paciencia, dedicación y recomendaciones brindadas en cada una de las etapas de investigación.

A mi tío Juan Carlos por ser un guía a lo largo de estos años de vida universitaria con sus directrices, consejos y conocimientos.

**Jhonatan Andrés**

## Índice de Contenidos

Dedicatoria .....	6
Agradecimientos.....	7
Índice de Contenidos.....	8
Índice de Tablas .....	13
Índice de Figuras.....	15
Resumen.....	21
Abstract .....	22
Capítulo I.....	23
Definición del Proyecto.....	23
Introducción .....	23
Antecedentes.....	25
Justificación e Importancia.....	27
Alcance.....	29
Objetivo General.....	30
Objetivos Específicos.....	30
Capítulo II.....	31
Marco Teórico .....	31
Internet of Things (IoT).....	31
Definición y arquitectura de <i>Smart Things</i> .....	34
Hidroponía.....	36
<i>Solución nutritiva</i> .....	38
<i>Smart Farming</i> .....	40

LPWAN.....	42
<i>Aplicaciones y Servicios Inteligentes</i> .....	43
Capítulo III.....	45
Materiales y Métodos .....	45
Comparación de tecnologías LPWAN.....	45
Análisis de la tecnología LPWAN LoRa .....	48
<i>LoRa</i> .....	48
<i>Radiofrecuencia</i> .....	53
<i>Capa física (PHY) de LoRa</i> .....	55
<i>Modulación Chirp Spread Spectrum</i> .....	57
Estándares.....	64
<i>LoRaWAN</i> .....	64
<i>MQTT</i> .....	69
Topologías de Red.....	73
<i>Topología de red estrella</i> .....	73
<i>Topología de red en árbol</i> .....	74
<i>Topología de red malla</i> .....	75
<i>Topología Repetidor</i> .....	77
<i>Topología de Enrutamiento</i> .....	78
<i>Elección de la topología de red</i> .....	80
Análisis de prestaciones técnicas para aplicaciones con LoRa.....	81
Módulos de LoRa.....	83
<i>LG01N LoRa Gateway</i> .....	83
<i>LoRa Shield</i> .....	85
Sensores para monitorización remota.....	86

	10
<i>Sensor de Temperatura</i> .....	87
<i>Sensor de Humedad</i> .....	89
<i>Sensor Ultrasónico</i> .....	90
<i>Sensor de pH</i> .....	92
Servidor en la nube.....	96
<i>Productos y Soluciones</i> .....	97
<i>Precios</i> .....	98
<i>Características</i> .....	100
<i>Elección del servidor en la nube</i> .....	101
Capítulo IV .....	102
Diseño e Implementación .....	102
Desarrollo del sistema hidropónico .....	102
Diseño del sistema hidropónico NFT.....	105
<i>Tuberías PVC</i> .....	106
<i>Selección del contenedor</i> .....	109
<i>Selección de la bomba</i> .....	111
<i>Solución nutritiva y plantación</i> .....	117
Arquitectura del sistema de monitorización .....	119
Diseño del Nodo final.....	120
<i>Diagrama de Conexiones</i> .....	120
<i>Configuración del software de la red de sensores</i> .....	122
<i>Calibración de los sensores</i> .....	124
<i>Calibración del sensor de pH</i> .....	126
<i>Configuración del sensor de temperatura</i> .....	130
<i>Configuración del sensor de humedad del suelo</i> .....	133

<i>Configuración del sensor de nivel</i> .....	136
<i>Programación del Nodo Final</i> .....	139
<i>Carcasa del Nodo Final</i> .....	144
<i>Frecuencia de Operación</i> .....	146
<i>Configuración RF de LoRa Shield</i> .....	146
<i>Configuración del Gateway LG01-N</i> .....	147
<i>Cálculo del área de cobertura</i> .....	148
Configuración del servidor virtual .....	153
Plataforma IoT .....	166
<i>Diseño Front End</i> .....	167
<i>Diseño Back End</i> .....	171
<i>Archivos de la Plataforma IoT</i> .....	172
<i>Comunicación Gateway LoRa-Bróker-BD</i> .....	173
<i>Configuración de Node JS</i> .....	174
<i>Comunicación Bróker-Plataforma IoT</i> .....	177
<i>Comunicación Plataforma IoT – BD</i> .....	179
<i>Configuración de la plataforma para el envío de alertas</i> .....	185
Implementación de la red de sensores.....	188
<i>Descripción del escenario</i> .....	188
<i>Localización geográfica</i> .....	190
<i>Implementación del nodo final</i> .....	191
Capítulo V .....	194
Pruebas y Resultados .....	194
Protocolo de pruebas.....	194
Análisis de Resultados.....	213

VARIABLES FÍSICAS.....	213
Semana 1 .....	217
Semana 2 .....	219
Semana 3 .....	221
Semana 4 .....	223
Parámetros .....	225
Conclusiones y Trabajos Futuros .....	231
Conclusiones .....	231
Trabajos Futuros.....	233
Referencias Bibliográficas .....	235

## Índice de Tablas

<b>Tabla 1</b> Funciones de los minerales en las plantas hidropónicas. ....	38
<b>Tabla 2</b> Características de medios de soporte en hidroponía .....	39
<b>Tabla 3</b> Aplicaciones de LPWAN.....	44
<b>Tabla 4</b> Comparativa de redes LPWAN.....	45
<b>Tabla 5</b> Comparación de Costos entre tecnologías LPWAN .....	46
<b>Tabla 6.</b> Relación entre Factor de Dispersión, Tasa de Bits, Sensibilidad y SNR.....	49
<b>Tabla 7</b> Relación entre SF, Throughput y ToA .....	52
<b>Tabla 8</b> Canales, frecuencias y subbandas de LoRa.....	54
<b>Tabla 9</b> Capacidades de detección y corrección de errores de LoRa.....	56
<b>Tabla 10</b> Comparación de Topologías de red.....	80
<b>Tabla 11</b> Características del enlace ascendente y descendente .....	81
<b>Tabla 12</b> Métricas de requisitos de aplicabilidad para aplicaciones IoT con LoRa.....	82
<b>Tabla 13</b> Especificaciones de consumo de energía.....	86
<b>Tabla 14</b> Sensores de Temperatura.....	87
<b>Tabla 15</b> Especificaciones Técnicas DHT11 .....	88
<b>Tabla 16</b> Especificaciones Técnicas SKU: SEN0114 .....	90
<b>Tabla 17</b> Especificaciones Técnicas HC-SR04 .....	92
<b>Tabla 18</b> Especificaciones Técnicas Sonda E201-BNC .....	94
<b>Tabla 19</b> Especificaciones Técnicas Tarjeta de Acondicionamiento 4502C .....	95
<b>Tabla 20</b> Soluciones de Cloud Computing .....	97
<b>Tabla 21</b> Comparativa de precios de Cloud Computing .....	100
<b>Tabla 22</b> Tipos de sistemas hidropónicos .....	103
<b>Tabla 23</b> Característica técnicas del tubo PVC .....	107
<b>Tabla 24</b> Características de la tubería.....	109
<b>Tabla 25</b> Parámetros de cálculo para la capacidad del contenedor.....	110

<b>Tabla 26</b> Dimensiones del contenedor .....	111
<b>Tabla 27</b> Parámetros para el cálculo de caudal.....	112
<b>Tabla 28</b> Parámetros técnicos bomba sumergible.....	116
<b>Tabla 29</b> Componentes de Soluciones Concentradas.....	117
<b>Tabla 30</b> Distribución de los pines de LoRa Shield .....	121
<b>Tabla 31</b> Componentes del software del Nodo Final .....	123
<b>Tabla 32</b> Parámetros de configuración en Radio Mobile .....	152
<b>Tabla 33</b> Security group .....	155
<b>Tabla 34</b> Configuración de dominio.....	159
<b>Tabla 35</b> Tecnologías utilizadas para el diseño Front End .....	168
<b>Tabla 36</b> Tecnologías utilizadas para la programación Back End.....	172
<b>Tabla 37</b> Dimensiones del invernadero .....	189
<b>Tabla 38</b> Ejemplo de decodificación del Payload .....	196

## Índice de Figuras

<b>Figura 1</b> Estructura de un Sistema IoT.....	33
<b>Figura 2</b> Diagrama de bloques de Smart Things.....	36
<b>Figura 3</b> Flujo de Información en Smart Farming.....	41
<b>Figura 4</b> Pila de Protocolos LoRa/LoRaWAN.....	48
<b>Figura 5</b> Relación SF, bitrate y energía.....	51
<b>Figura 6</b> Canales para Europa, Estados Unidos y Australia.....	53
<b>Figura 7</b> Arquitectura de la capa PHY de LoRa.....	55
<b>Figura 8</b> Espectrograma de los factores de propagación de LoRa y CSS.....	58
<b>Figura 9</b> Señal Upchirp y Downchirp lineal.....	59
<b>Figura 10</b> Señales Upchirp y Downchirp en frecuencia.....	60
<b>Figura 11</b> Modulación de la señal.....	60
<b>Figura 12</b> Señal LoRa.....	62
<b>Figura 13</b> Formato de la Trama Física de LoRa.....	63
<b>Figura 14</b> Mensaje codificado en las señales de chirp.....	64
<b>Figura 15</b> LoRaWAN Clase A.....	66
<b>Figura 16</b> LoRaWAN Clase B.....	66
<b>Figura 17</b> LoRaWAN Clase C.....	67
<b>Figura 18</b> Elementos de la red LoRaWAN.....	68
<b>Figura 19</b> Protocolo MQTT.....	69
<b>Figura 20</b> Diagrama Publish-Subscribe.....	70
<b>Figura 21</b> Nomenclatura de los Tópicos.....	71
<b>Figura 22</b> Esquema Publish, Subscribe y Bróker.....	72
<b>Figura 23</b> Red con topología estrella.....	74
<b>Figura 24</b> Red con topología en árbol.....	75
<b>Figura 25</b> Red con topología de malla.....	76

<b>Figura 26</b> Topologías con nodos relay .....	78
<b>Figura 27</b> Topologías con nodos router .....	79
<b>Figura 28</b> Dragino LG01 Gateway .....	84
<b>Figura 29</b> Dragino LoRa Shield.....	85
<b>Figura 30.</b> Sensor DHT11 .....	88
<b>Figura 31</b> Sensor de Humedad SKU: SEN0114.....	89
<b>Figura 32</b> Sensor Ultrasónico HC-SR04 .....	91
<b>Figura 33.</b> Estructura para la medición de pH.....	93
<b>Figura 34</b> Sonda de pH E201-BNC.....	94
<b>Figura 35</b> Tarjeta de acondicionamiento PH-4502C.....	95
<b>Figura 36</b> Tipos de sistemas hidropónicos.....	102
<b>Figura 37</b> Diagrama del sistema hidropónico NFT .....	104
<b>Figura 38</b> Sistema hidropónico NFT y red de sensores para monitorización remota...	105
<b>Figura 39</b> Tubería con protección UV .....	108
<b>Figura 40</b> Estructura circular de la tubería .....	108
<b>Figura 41</b> Tubo circular.....	112
<b>Figura 42</b> Diseño del sistema hidropónico .....	114
<b>Figura 43</b> Selección de bomba sumergible .....	115
<b>Figura 44</b> Bomba sumergible.....	116
<b>Figura 45</b> Siembra de semillas.....	118
<b>Figura 46</b> Trasplante de semillas.....	118
<b>Figura 47</b> Diagrama de bloques del sistema de monitorización .....	119
<b>Figura 48</b> Diagrama físico del Nodo Final .....	122
<b>Figura 49</b> Equipos patrones.....	125
<b>Figura 50</b> Proceso de calibración del sensor de pH .....	126
<b>Figura 51</b> Diagrama de flujo para configuración del sensor de pH.....	127

<b>Figura 52</b> Mediciones de pH con equipo patrón.....	128
<b>Figura 53</b> Curva de calibración sensor de pH.....	129
<b>Figura 54</b> Diagrama de dispersión mediciones de pH.....	130
<b>Figura 55</b> Curva de calibración sensor de temperatura.....	132
<b>Figura 56</b> Diagrama de dispersión mediciones de temperatura .....	133
<b>Figura 57</b> Curva de calibración sensor de humedad.....	135
<b>Figura 58</b> Diagrama de dispersión mediciones de temperatura .....	136
<b>Figura 59</b> Diagrama de flujo para el sensor de nivel .....	137
<b>Figura 60</b> Curva de calibración sensor de humedad.....	138
<b>Figura 61</b> Diagrama de dispersión mediciones de temperatura .....	139
<b>Figura 62</b> Carga útil formado por las variables de monitorización .....	140
<b>Figura 63</b> Diagrama de flujo del sistema de monitorización .....	141
<b>Figura 64</b> Configuración del Nodo Final.....	142
<b>Figura 65</b> Carcasa del Nodo Final .....	145
<b>Figura 66</b> Configuración de RF.....	146
<b>Figura 67</b> Credenciales de sesión Dragino .....	147
<b>Figura 68</b> Configuración de parámetros LoRa en el Gateway.....	147
<b>Figura 69</b> Configuración MQTT en el Gateway.....	148
<b>Figura 70</b> Presupuesto del enlace .....	149
<b>Figura 71</b> Simulación del área de cobertura .....	153
<b>Figura 72</b> Selección de la instancia .....	154
<b>Figura 73</b> Configuración del almacenamiento.....	155
<b>Figura 74</b> Llave de la instancia .....	156
<b>Figura 75</b> Conversión del archivo .pem.....	157
<b>Figura 76</b> Conexión vía SSH a la instancia.....	158
<b>Figura 77</b> Plataforma IoT sin certificados SSL .....	159

<b>Figura 78</b> Instalación de certificados SSL .....	160
<b>Figura 79</b> Certificados SSL .....	161
<b>Figura 80</b> Plataforma IoT con certificados SSL .....	161
<b>Figura 81</b> Configuración de EMQ.....	163
<b>Figura 82</b> Base de datos.....	163
<b>Figura 83</b> Tabla de datos .....	164
<b>Figura 84</b> Pestaña Remote de configuración FTP.....	165
<b>Figura 85</b> Conexión FTP.....	166
<b>Figura 86</b> Esquema de tecnologías .....	167
<b>Figura 87</b> Estructura básica Front End.....	169
<b>Figura 88</b> Estructura de la plataforma IoT.....	170
<b>Figura 89</b> Plataforma IoT .....	171
<b>Figura 90</b> Diagrama de archivos de la Plataforma IoT .....	173
<b>Figura 91</b> Esquema de transmisión y almacenamiento de datos en la BD.....	174
<b>Figura 92</b> Versiones de Node JS y NPM.....	174
<b>Figura 93</b> Instalación de componentes para Node JS.....	175
<b>Figura 94</b> MQTT y WebSockets.....	178
<b>Figura 95</b> Diagrama comunicación entre usuario y servidor.....	180
<b>Figura 96</b> Proceso almacenado para consulta de históricos .....	184
<b>Figura 97</b> Procedimientos almacenados .....	185
<b>Figura 98</b> Información del servicio .....	186
<b>Figura 99</b> Manejador de dependencias.....	187
<b>Figura 100</b> Instalación de nodo Twilio.....	187
<b>Figura 101</b> Vista externa del invernadero .....	190
<b>Figura 102</b> Localización geográfica .....	191
<b>Figura 103</b> Nodo Final .....	192

<b>Figura 104</b> Implementación del Nodo Final.....	193
<b>Figura 105</b> Escenario adquisición de datos .....	194
<b>Figura 106</b> Adquisición de datos.....	195
<b>Figura 107</b> Esquema de verificación de recepción.....	195
<b>Figura 108</b> Conexión vía SSH al Gateway .....	196
<b>Figura 109</b> Paquete de datos recibido .....	196
<b>Figura 110</b> Red WiFi.....	197
<b>Figura 111</b> Petición ICMP al Gateway LoRa.....	198
<b>Figura 112</b> Respuesta ICMP Gateway Servidor.....	199
<b>Figura 113</b> Tracert dirigido al servidor "wirelbits.ga" .....	200
<b>Figura 114</b> Escenario transmisión de los datos.....	200
<b>Figura 115</b> LogRead de transmisión de datos al servidor .....	201
<b>Figura 116</b> Configuración de WebSocket .....	202
<b>Figura 117</b> Llegadas de mensajes al bróker .....	203
<b>Figura 118</b> Visualización de los datos en tiempo real .....	203
<b>Figura 119</b> Envío de mensajes desde la plataforma .....	204
<b>Figura 120</b> LogRead del Gateway LoRa .....	205
<b>Figura 121</b> Escenario de verificación de almacenamiento .....	205
<b>Figura 122</b> Logs de Node.JS .....	206
<b>Figura 123</b> Inserción de los datos en la BD .....	206
<b>Figura 124</b> Estadísticas en tiempo real de los datos .....	207
<b>Figura 125</b> Gadgets de visualización .....	208
<b>Figura 126</b> Interfaz gráfica de históricos .....	209
<b>Figura 127</b> Datos históricos de las variables físicas.....	210
<b>Figura 128</b> Formulario de SetPoints .....	211
<b>Figura 129</b> Inserción de los SetPoints.....	212

<b>Figura 130</b> Logs de las alertas SMS enviadas .....	212
<b>Figura 131</b> Señales físicas obtenidas semana 1 .....	214
<b>Figura 132</b> Señales físicas obtenidas semana 2 .....	215
<b>Figura 133</b> Señales físicas obtenidas semana 3 .....	216
<b>Figura 134</b> Señales físicas obtenidas semana 4 .....	217
<b>Figura 135</b> Proceso de crecimiento de las plantas .....	218
<b>Figura 136</b> BoxPlot Variables Físicas - Semana 1 .....	219
<b>Figura 137</b> Proceso de crecimiento de las plantas .....	220
<b>Figura 138</b> BoxPlot Variables Físicas - Semana 2 .....	221
<b>Figura 139</b> Proceso de crecimiento de las plantas .....	222
<b>Figura 140</b> BoxPlot Variables Físicas - Semana 3 .....	223
<b>Figura 141</b> Proceso de crecimiento de las plantas .....	224
<b>Figura 142</b> BoxPlot Variables Físicas - Semana 4 .....	225
<b>Figura 143</b> SNR y RSSSI obtenidos - Semana 1 y 2 .....	227
<b>Figura 144</b> SNR y RSSSI obtenidos - Semana 3 y 4 .....	228
<b>Figura 145</b> Retardos .....	229
<b>Figura 146</b> Paquetes perdidos .....	230
<b>Figura 147</b> Paquetes Perdidos .....	230

## Resumen

En los últimos años, las masivas aplicaciones de IoT (del inglés *Internet of Things*) implementadas para la monitorización de cultivos con sensores inalámbricos han acelerado la evolución de la agricultura tradicional y combinadas con tecnologías de la información para realizar algoritmos de predicción la han convertido en inteligente (*Smart Farming*). En el presente Proyecto de Investigación se aplicaron los beneficios de la tecnología de IoT para desarrollar una plataforma de bajo costo, basada en la red LoRa y orientada al *Smart Farming*, destinada a mejorar la gestión y monitorización de cultivos hidropónicos. El sistema de monitorización está compuesto de un nodo inteligente instalado en el sistema hidropónico, el mismo que contiene sensores que permitieron medir variables físicas como el pH, humedad, temperatura y nivel del agua, se proporcionó al usuario una plataforma de visualización basada en la *Web* para tener un reporte detallado de los valores adquiridos mediante el nodo, controlar los elementos de salida mediante actuadores como relés, establecer rangos de operación de los sensores y recibir alertas en tiempo real vía SMS. Como resultado de la implementación, fue posible determinar que para una producción de 40 plantas el consumo de agua es progresivo pues aumenta de 8.5 litros a 14 litros, los rangos de pH de 6 y 6.5 de la solución nutritiva permiten que las plantas crezcan y maduren apropiadamente, los niveles de humedad superiores al 40% y temperaturas con valores entre 5 a 25 grados hizo posible la pronta germinación de las semillas en 21 días.

### **PALABRAS CLAVE:**

- **IoT**
- **LoRa**
- **SMART FARMING**
- **HIDROPONÍA**

### **Abstract**

In recent years, massive IoT (Internet of Things) applications implemented for crop monitoring with wireless sensors have accelerated the evolution of traditional agriculture and combined with information technologies for predictive algorithms have turned it into Smart Farming. In the present research project, the benefits of IoT technology were applied to develop a low-cost platform, based on the LoRa network and oriented to Smart Farming, aimed at improving the management and monitoring of hydroponic crops. The monitoring system is composed of a smart node installed in the hydroponic system, the same that contains sensors that allowed measuring physical variables such as pH, humidity, temperature and water level, the user was provided with a web-based visualization platform to have a detailed report of the values by through the node, control the output elements by actuators such as relays, set sensor operation ranges and receive real-time alerts via SMS. As a result of the implementation, it was possible to determine that for a production of 40 plants the water consumption is progressive as it increases from 8.5 liters to 14 liters, the pH ranges of 6 and 6.5 of the nutrient solution allow the plants to grow and mature properly, the humidity levels above 40% and temperatures with values between 5 to 25 degrees made possible the prompt germination of the seeds in 21 days.

### **KEYWORDS:**

- **IoT**
- **LoRa**
- **SMART FARMING**
- **HYDROPONICS**

## Capítulo I

### Definición del Proyecto

#### Introducción

La constante investigación en las últimas décadas ha originado un desarrollo sustancial en ámbitos de tecnologías de comunicación inalámbrica, ciencias de la información y comunicación, sistemas industriales y electromecánicos, lo que ha impulsado la evolución de una nueva tecnología conocida como Internet de las Cosas (*IoT*, del inglés *Internet of Things*). El IoT es uno de los paradigmas emergentes en los últimos años debido al gran impacto social, técnico y económico (Peng, Pal, & Huang, 2020). El propósito más importante del IoT es tener la capacidad de realizar la conexión de todos o cualquier dispositivo a la *Internet* o entre otros dispositivos conectados. La red de manera general está compuesta por diversos aparatos de uso doméstico, dispositivos físicos, redes de vehículos y máquinas equipadas con sensores, componentes electrónicos y actuadores, que combinados con la interconexión de la red hace factible el proceso de recolección, almacenamiento e intercambio de datos/información.

Para el año 2025 se prevé que existan unos 70.000 millones de dispositivos conectados. Se estima que el impacto siga en crecimiento debido a la convergencia de la red, combinación con otras tecnologías como inteligencia artificial, *machine learning*, *deep learning*, *big data* y el 5G. La transformación tecnológica generada por el IoT, involucra una serie amplia de cambios de infraestructura, rápido crecimiento, prevalencia de servicios tecnológicos en la nube para el almacenamiento y ejecución de algoritmos, disminución del tamaño de los dispositivos y aceleración de las funcionalidades informáticas de los equipos.

Con esta transformación surgen diversos ámbitos de aplicaciones empresariales, comerciales e industriales para aplicar soluciones con IoT, uno de estos campos es la agricultura de precisión (*Smart Farming*). Con la implementación del IoT en los procesos agrícolas se fortalece el trabajo y se incrementan los niveles de producción debido a la monitorización minuciosa de variables como temperatura, humedad y pH para colocar las dosis adecuadas de solución nutritiva que necesitan las plantas para su óptimo crecimiento. Con el uso de sensores inteligentes es posible examinar las condiciones meteorológicas en los sembríos y predecir enfermedades ocasionadas por excesivas lluvias y viento. Los nodos finales colocados en los suministros de agua inspeccionan su estado para saber si puede ser utilizada para el regadío o recirculación en sistemas hidropónicos. Los diferentes dispositivos inteligentes que forman parte de la red IoT, tienen docenas de sensores y actuadores conectados a ellos, lo que se pretende es tener la capacidad de transmitir y recibir datos en tiempo real, sin embargo, existen diferentes problemas de red como la poca fiabilidad de las comunicaciones inalámbricas y ciertos entornos con latencia alta (Latief & Firasath, 2021).

Algunos nodos finales presentan limitaciones en potencia, alimentación y bajos recursos. El uso del ancho de banda es uno de los aspectos primordiales dado que algunos dispositivos presentan conexiones limitadas (Hillar, 2017), por estas razones para poner en ejecución proyectos con esta tecnología es indispensable disponer de amplias áreas de cobertura, con el propósito de transmitir los datos extraídos de la monitorización de variables físicas de los sensores desde lugares remotos hacia los centros de procesamiento/almacenamiento de información, clientes y consumidores. Diversos estándares para transmisión de larga distancia han sido desarrollados, e incluyen los de las redes celulares como 4G LTE y 5G, no obstante, existen otras tecnologías de largo alcance como LoRa, Sigfox, Telensa y Multefire. Finalmente, los

datos transmitidos al utilizar estos estándares son analizados con herramientas estadísticas, técnicas de inteligencia artificial, aprendizaje automático y en la nube mediante motores de reglas y algoritmos.

### **Antecedentes**

Aproximadamente, más de 50.000 millones de dispositivos se encontrarán interconectados en 2022 con el uso de comunicaciones por radio. Con el acelerado crecimiento del IoT en el mercado de las telecomunicaciones, las redes de área amplia de baja potencia (*LPWAN*, del inglés *Low Power Wide Area Networks*) emergen como una tecnología de radiocomunicación de largo alcance. Las principales tecnologías que se encuentran en competición por su despliegue en redes IoT a gran escala son: Sigfox, LoRa y NB-IoT. Una investigación comparativa de estas tecnologías se plantea en (Mekki, Bajic, Chaxel, & Meyer, *A Comparative Study of LPWAN Technologies for Large-scale IoT Deployment*, 2017) en la cual las erige como soluciones de alta eficiencia para la conexión entre dispositivos inteligentes, autónomos y heterogéneos. En términos de consumo de energía, capacidad y costos la investigación pone en manifiesto las ventajas de LoRa y Sigfox, mientras que, NB-IoT dispone de mejores prestaciones respecto a calidad de servicio y latencia. Un enfoque sistemático se realiza en (Chaudhari, Zennaro, & Borkar, *LPWAN Technologies: Emerging Application Characteristics, Requirements, and Design Considerations*, 2020), donde se establecen a las redes LPWAN como una solución innovadora para transmisiones de largo alcance y baja potencia en aplicaciones del IoT y *Machine to Machine (M2M)* donde se analizan características principales de estas redes como: consumo de energía, área de cobertura, capacidad y costos del despliegue de la red y equipos. Así mismo, como en cualquier tipo de red antes del despliegue con LPWAN se deben considerar aspectos de diseño, entre las más importantes están: la gestión del tráfico de usuarios, gestión de

interferencias, modos de funcionamiento para ahorrar energía, protocolos de control de acceso al medio (MAC, del inglés *Media Access Control*) ligeros, técnicas de seguridad, entre otras.

Las múltiples aplicaciones de IoT y LPWAN permiten solventar problemas en procesos agrícolas originados por el escaso uso de tecnología y la alta demanda de producción suscitada por el crecimiento acelerado de la población. Con la adopción de tecnología de comunicación de bajo costo y largo alcance en la agricultura, aparece el término de *Smart Farming*. La agricultura inteligente o de precisión busca garantizar el cumplimiento adecuado de suministro de alimentos, estabilidad y sostenibilidad. En la investigación de (Citoni, Imran, & Abassi, 2019) se muestra mediante el uso de IoT en sistemas de riego, se optimiza el consumo de agua, el uso de energía eléctrica y la reducción de mano de obra. El bajo costo y fácil uso son requerimientos fundamentales para que la agricultura inteligente sea viable, con el uso de sensores en plantaciones, suelos y sistemas hidropónicos se perfecciona el proceso de suministro de nutrientes y crecimiento de las plantas.

Una gran cantidad de sensores se localizan en la capa física, cuya funcionalidad consiste en la monitorización de diversas variables y parámetros indispensables para el presente proyecto. En el mercado existen innumerables sensores que pueden emplearse para monitorizar temperatura, humedad, pH y nivel, por ejemplo, en (Davcev, Mitreski, Trajkovic, Nikolovski, & Koteli, 2019) se expone un sistema agrícola innovador compuesto de nodos sensores como el DTH22 para monitorizar humedad y temperatura, el sensor ultrasónico para medir el nivel de agua en las soluciones nutritivas en cultivos de uva. Posteriormente emplean la red LoRaWAN para la comunicación y transmisión a larga distancia de los datos recopilados por los sensores y la plataforma de *The Things Network* (TTN), que es altamente escalable para el

almacenamiento y procesamiento en la nube con fines estadísticos, procesos de gestión y control de los nodos terminales de la red. El servicio de control remoto permite emitir alertas en tiempo real que pueden ser automatizados por el servicio de análisis de datos o por el usuario desde la plataforma web.

### **Justificación e Importancia**

El área agrícola en nuestro país presenta muchos problemas en la optimización de los procesos de producción ya que durante muchos años este sector ha estado en profundo estancamiento, los apoyos gubernamentales son muy pocos, lo que desmotiva invertir en el campo, además de las graves pérdidas económicas generadas por aspectos naturales como heladas, plagas de insectos y aves, lluvias intensas y sequías de larga duración.

Ecuador es un país con una producción netamente agrícola, se encuentra ubicado entre los países de Sudamérica en el cual todavía se conservan las técnicas de agricultura tradicional que ocasiona sistemas económicos agrarios con bajos niveles de producción. El sector agrícola es el generador primordial de empleo en el país, así como también del 8% del Producto Interno Bruto (PIB). No obstante, la inexistente aplicación de nuevas metodologías de cultivo provoca la explotación de mano de obra, tendencia al monocultivo, erosión del suelo y estancamiento económico.

La población ecuatoriana depende en gran medida de la producción de cereales, frutas, verduras, hortalizas y leche, lo que convierte al país como un gran potencial para la producción agrícola a nivel mundial. Si bien, aún existen limitaciones con respecto a la productividad, enfoques, inversión y zonificación del sector agrícola, para superar esas carencias se debe adoptar, promover y realizar investigación ardua de los

beneficios de la utilización de nuevas técnicas agrícolas y combinación con otras tecnologías y ciencias.

Para sustituir los mecanismos tradicionales de agricultura y volverla sostenible se han introducido innovadores procedimientos, conceptos, metodologías y sobre todo tecnología. Una de estos procedimientos es la hidroponía, un sistema agrícola altamente eficaz, que se emplea para el cultivo de inmensos tipos de plantas, frutas y verduras. Los sistemas hidropónicos son altamente adaptables a cualquier entorno debido a su independencia a la necesidad de tierra, lo que la ha convertido en una práctica cada vez más popular. La mayor ventaja que presenta la hidroponía al ser un entorno sin suelo es que compensa las desventajas existentes con el uso de agricultura en tierra al evitar enfermedades en los cultivos por el contacto con el suelo, la utilización de químicos, pesticidas y principalmente daños por perturbaciones climáticas.

También se posibilita a los productores obtener alimentos en cualquier parte de sus hogares, en cualquier época del año, y obtener mayores rendimientos con menos recursos. Adicionalmente, con el uso de sensores que transmiten información en tiempo real e instalados de manera adecuada en el contenedor de la solución nutritiva y agua del sistema hidropónico, se puede realizar una monitorización precisa de variables como el pH, la humedad, temperatura y el nivel de agua. Monitorizar estos parámetros es de suma importancia para garantizar que las plantas obtengan los nutrientes exactos que necesitan para un desarrollo óptimo, minimizar la muerte en los cultivos ocasionados por el exceso o escasez de los mismos, disminuir el tiempo de cosecha y costos de producción.

La importancia de desarrollar este tipo de sistemas es demostrar en base a un prototipo que se puede implementar tecnología inalámbrica a cultivos hidropónicos y monitorizar zonas más extensas mediante varios nodos y LPWAN. En este caso el

prototipo permite monitorizar con un solo nodo final al cual están conectados varios sensores los parámetros básicos del sistema hidropónico, para la transmisión se usa tecnología de espectro ensanchado que permite velocidades de datos que no interfieren con otras comunicaciones. Para la implementación de la red de sensores inalámbricos se hace uso de la tecnología LoRa debido a las grandes ventajas que ofrece. El atractivo comercial de LoRa se debe a que un sólo Gateway y diferentes protocolos de comunicación se puede transmitir los datos a un servidor en la nube.

### **Alcance**

En el presente proyecto de investigación se aplican los beneficios de la tecnología de IoT con el propósito de desarrollar una plataforma de bajo costo, basada en la red LoRa y orientada al *Smart Farming* con la intención de mejorar la gestión y monitorización de cultivos hidropónicos. El sistema de monitorización está compuesto de un nodo ubicado en el sistema hidropónico, el mismo que contiene sensores que permiten medir variables físicas como el pH, humedad, temperatura y nivel del agua. Los sensores de humedad y temperatura se encuentran ubicados en el semillero de germinación de plantas de tomate, mientras que el sensor de pH y ultrasónico en el contenedor de la solución nutritiva que contiene nitrógeno, fósforo, magnesio y calcio. Los sensores poseen una conexión de tipo alámbrica hacia la *Shield* de *LoRa – Arduino* que realiza la comunicación mediante *LoRa* con el *Gateway* previamente realizada la configuración de parámetros de la capa física. Posteriormente, se requiere intercambiar datos entre dispositivos casi en tiempo real por medio de la *Internet* y consumir el menor ancho de banda posible por lo cual se hace uso del protocolo de mensajería ligera MQTT (del inglés, *Message Queing Telemetry Transport*), que funciona de manera eficiente en redes inalámbricas y que permite recopilar los datos sensados en un servidor creado en la plataforma de computación en la nube *Amazon Web Services*, con

el propósito de brindar la posibilidad de almacenar toda la información en una base de datos para que pueda ser analizada. Finalmente, se proporciona al usuario una herramienta de visualización basada en la WEB para tener un reporte más detallado de los valores adquiridos mediante los equipos con gráficas y *gadgets*, controlar el elemento de salida (actuador): relé de la bomba encargada de la circulación de agua, establecer rangos de operación de los sensores (*Set-Points*) y recibir alertas en tiempo real vía SMS.

### **Objetivo General**

Implementar una red de sensores inalámbricos para la monitorización de variables físicas en cultivos hidropónicos mediante tecnología LoRa.

### **Objetivos Específicos**

- Analizar las características, prestaciones y topologías de red que presenta la tecnología LoRa y los estándares requeridos para el desarrollo de la red de sensores inalámbrica.
- Diseñar la red de sensores inalámbricos para la monitorización en tiempo real de variables físicas en cultivos hidropónicos.
- Desplegar la red de sensores inalámbricos en el sistema hidropónico para adquisición y procesamiento de la información.
- Desarrollar una interfaz gráfica orientada a la Web que permita al usuario visualizar en tiempo real las variables físicas del cultivo hidropónico monitorizado.
- Evaluar el desempeño de la red inalámbrica de sensores para el análisis de los parámetros asociados a la de Calidad de Servicio (QoS).

## Capítulo II

### Marco Teórico

#### Internet of Things (IoT)

Este término fue utilizado por primera vez en el año de 1998 por Kevin Ashton informático británico de Procter & Gamble. IoT surgió como un nuevo paradigma que permitiera solucionar los problemas de comunicación para establecer el tiempo y lugar en el cual los productos más populares de Procter & Gamble se agotaban en las diversas tiendas de Inglaterra. Ashton propuso una idea revolucionaria que consistía en colocar una serie de sensores en los productos que permitiera realizar un seguimiento y establecer de una manera más eficiente, cuándo y dónde se habían vendido todos los productos de la empresa. Sin embargo, el paradigma de IoT arrancó sino hasta el año 2009 y se definió como “la tercera ola revolucionaria de la tecnología de la información”, debido a que con el uso de IoT es posible recolectar, procesar y analizar datos que permiten la generación de algoritmos y nuevos modelos de predicción (Cardona, Vijender, & Cecilia, 2021).

IoT presenta diversas definiciones, según Milenkovic (Milan, 2020) “constituye un sistema que posibilita la conexión de objetos del mundo físico a Internet, entrega diversas herramientas tecnológicas para monitorizar, procesar y automatizar variables físicas”, la Unión Internacional de Telecomunicaciones (*UIT*) en su informe sobre ciudades inteligentes y sostenibles en el año 2015, define al IoT como “una estructura de información de carácter mundial, que posibilita servicios tecnológicos por medio de la conexión de objetos tanto físicos y virtuales y que están basados en diversas herramientas informáticas y de telecomunicaciones existentes y en procesos de investigación” (Qusay F, Khan, & Sajjad A, 2017), y para IBM (Farshad & Krishnendu, 2020), "IoT es la capacidad de interconectar cualquier objeto del mundo físico a Internet,

permite la recolección de una gran cantidad de datos cada día que luego serán correlacionados y analizados para la automatización de decisiones eficientes e inteligentes, de esta manera se elimina la brecha existente entre la parte real y lo virtual/informático".

En términos generales el IoT se puede definir como una red extensa de interconexión entre objetos inteligentes (físicos o virtuales) que interactúan entre sí, que tienen la capacidad de compartir datos con el uso de tecnologías de la información e interoperación con la infraestructura de Internet y deriva en la capacidad de supervisar y controlar una multitud de parámetros físicos de forma remota y eficiente.

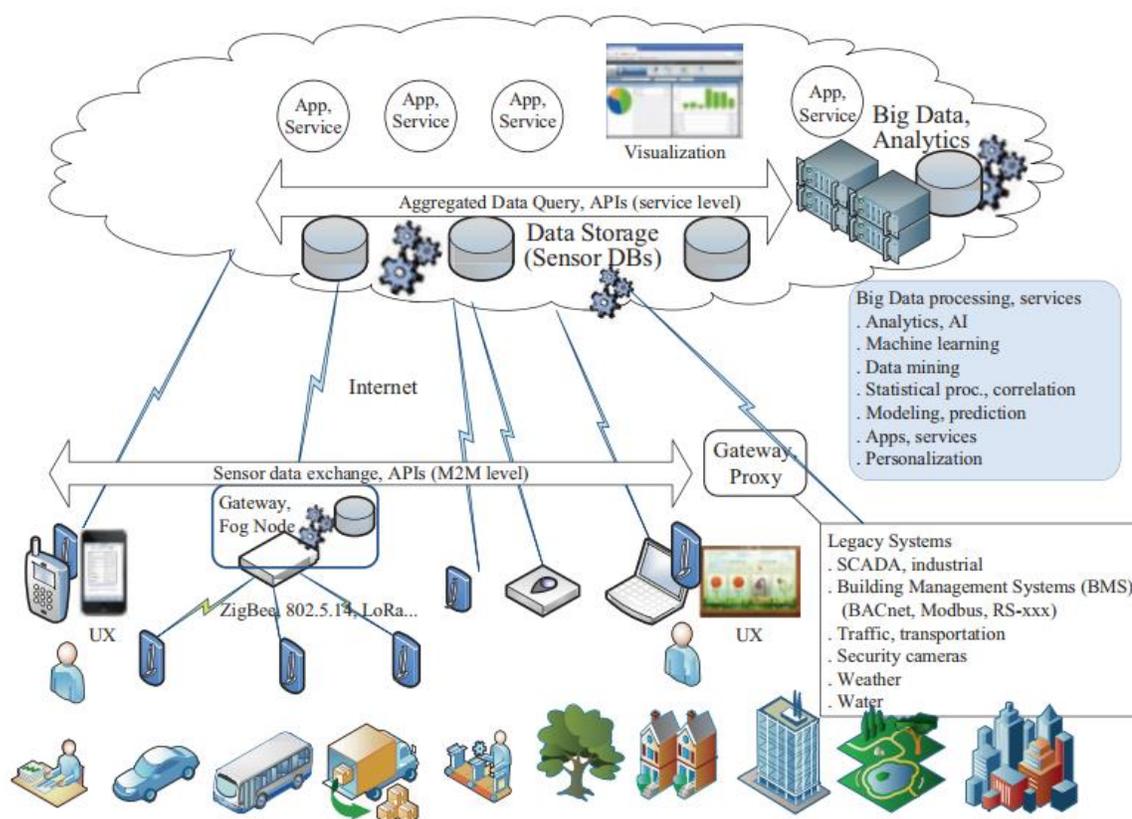
La Figura 1 presenta los elementos funcionales de IoT, la base constituye un sinnúmero de aplicaciones donde se puede aplicar este sistema: agricultura, oficinas, ciudades, retail, transporte, etc. El siguiente nivel está representado por los sensores y diversos objetos que permite la recolección de información. Algunos sensores y dispositivos frecuentemente denotados como inteligentes, tales como cámaras de seguridad, sensores de incendios, medidores de energía, termostatos presentan capacidades para conectarse de una manera directa hacia Internet e interactuar con diversos servicios y aplicaciones que se encuentran ejecutándose en la nube (Milan, 2020).

Otros sensores realizan la comunicación con el resto del sistema IoT e Internet mediante dispositivos intermedios conocidos como *Gateways*. Los *Gateways* y nodos son dispositivos que presentan mayor potencia y que se conectan a diferentes tipos de sensores normalmente de gama baja por medio de enlaces de red local y comúnmente de manera inalámbrica (Milan, 2020)

De manera conceptual, la nube (servidores remotos) constituye la capa superior en el sistema de IoT, en la que se realizan los procesos de *back-end* y algunos servicios comunes de tecnologías de la información (*TI*) como el proceso de almacenar información a gran escala, ejecución de motores para el procesamiento de análisis, aplicaciones para visualizar datos mediante gráficas y estadísticas, así como también de metodologías de seguridad. En algunos casos en la nube suelen estar en funcionamiento algoritmos de aprendizaje automático (*ML* del inglés, *machine learning*) y de inteligencia Artificial (*AI*, del inglés *Artificial Intelligence*) que mejoran su entrenamiento y capacidad de predicción al trabajar con la agregación constante de datos (Milan, 2020).

### Figura 1

#### Estructura de un Sistema IoT



*Nota.* El gráfico representa los componentes funcionales que conforman a un sistema IoT y la forma en que interactúan cada uno de ellos. Tomado de *IoT systems* (p.2) por M, Milenkovic, 2020, Springer.

### **Definición y arquitectura de *Smart Things***

El término *Smart Things* está relacionada a expresiones similares como objeto inteligente, dispositivo IoT, dispositivo inteligente, punto final IoT o dispositivo final. Según (Farshad & Krishnendu, 2020) *Smart Things*, independientemente de la expresión utilizada “es un objeto/dispositivo integrado con electrónica que posee la facultad para intercambiar datos por medio de una red sin ninguna interacción humana”. Como se presenta en la Figura 2; la unidad de procesamiento, dispositivos de comunicación, fuente de energía, sensores y actuadores son los componentes que forman un objeto inteligente.

- **Unidad de Procesamiento:** tiene la función de adquirir datos, procesar y analizar la información recopilada por los sensores, se encarga de sincronizar las diferentes señales de control de los actuadores y ejerce cierto dominio sobre algunos componentes de los objetos inteligentes como las fuentes de comunicación y alimentación. Existen diferentes unidades de procesamiento que pueden ser utilizadas según el tipo de aplicación y necesidad, no obstante, el microcontrolador presenta un uso más frecuente debido a su tamaño, fácil nivel de programación, flexibilidad, consumo de energía eficiente y bajo costo (Farshad & Krishnendu, 2020).
- **Transductores:** se definen como dispositivos que pueden convertir/transferir un tipo de energía en otra. Los campos de energía más comunes son el eléctrico, térmico, químico, de fluidos y mecánico. Los transductores se clasifican en: sensores y actuadores. Tanto actuadores como sensores propician a los objetos

inteligentes realizar la interacción con el mundo real (Farshad & Krishnendu, 2020).

- **Sensores:** reconocen la existencia de cambios y transferencias de energía tales como: el calor, luz, movimiento o reacciones químicas y luego generan una señal que pueda ser leída o procesada. Comúnmente, la salida es una señal de tipo eléctrica (digital o analógica) (Farshad & Krishnendu, 2020).
- **Actuadores:** son dispositivos que mediante el uso de energía realizan la producción de movimiento, transforman la energía mecánica en movimiento. Existen dos tipos de movimiento básicos: movimiento lineal y movimiento rotatorio (Farshad & Krishnendu, 2020).
- **Dispositivo de Comunicación:** permite la conexión de un objeto inteligente con el mundo real o con otros dispositivos por medio de la red, con el uso cables o de manera inalámbrica. Al emplear una red *IoT*, en la mayoría de casos los objetos inteligentes realizan una conexión de tipo inalámbrica dadas las limitaciones en infraestructura, mayor facilidad de despliegue y rentabilidad (Farshad & Krishnendu, 2020).
- **Fuente de Energía:** para su funcionamiento los objetos inteligentes necesitan energía, el dispositivo de comunicación es el componente que más energía consume, es por ello que es recomendable diseñar un objeto inteligente con uso de energía eficiente, componentes de bajo consumo y modos de suspensión (Farshad & Krishnendu, 2020).

## Figura 2

Diagrama de bloques de Smart Things



*Nota.* La figura expone mediante diagramas de bloques los componentes principales de un objeto inteligente. Tomado de *Typical smart things in IoT* (p.52) por F, Firouzi, 2020, Springer.

## Hidroponía

Según (Mohammed, 2018), la hidroponía “es la ciencia de cultivar plantas sin emplear la tierra, trabajo con agua”, proviene de las palabras griegas "*hydro*" y "*ponos*" que significa "agua" y "trabajo" respectivamente. La hidroponía es un estilo de agricultura que no requiere tierra sino de una solución nutritiva enriquecida con agua y minerales que son esenciales para el crecimiento óptimo de las plantas. Para ello se precisa de un correcto suministro de agua con un contenido bajo de partes por millón (*ppm*) y que los niveles de sal de potencial hidrógeno (*pH*) estén comprendidos entre 5.5-6.5. Un sistema hidropónico eficiente consiste en distribuir oportunamente los nutrientes esenciales para que las plantas crezcan en condiciones adecuadas (Mohammed, 2018).

A continuación, se definen algunos términos importantes para comprender un cultivo hidropónico:

- **Fotosíntesis:** es el modo en que las plantas realizan el proceso de síntesis de compuestos orgánicos en función de materia prima inorgánica con la utilización de la luz solar. Es el proceso de transformación de energía luminosa en energía química y posterior almacenamiento en enlaces de azúcar, es la clave para un crecimiento adecuado y alta productividad, debido a que las plantas convierten el dióxido de carbono y el agua en carbohidratos y oxígeno (Mohammed, 2018)
- **pH:** nivel de intensidad de la alcalinidad o acidez de una sustancia. Define la actividad de los iones de hidrógeno en una solución. Para medir los niveles de pH se emplea escala logarítmica de 0 hasta 14. Por medio del valor de pH se determinan las propiedades químicas del suelo o agua, entre ellas: de 0-6 presenta propiedades ácidas, 7-8 propiedades neutras y de 8-14 propiedades alcalinas (Mohammed, 2018).

Se miden los valores de pH en las soluciones nutritivas para cultivos hidropónicos con el propósito de definir la concentración relativa de iones de hidrógenos positivos, en vista de que las plantas se nutren al producirse el intercambio de iones. De este modo se puede determinar qué nutrientes necesita la planta o el cultivo (Mohammed, 2018).

- **Conductividad eléctrica (C.E.):** medida de la concentración salina o conductividad eléctrica de una solución o agua. Establece la sal disuelta en una solución con una escala que refleja los nutrientes presentes en la misma. La conductividad eléctrica se mide en mili Siemens por cm [mS/cm]. Por medio de la conductividad eléctrica es posible definir los nutrientes de la solución nutritiva (Mohammed, 2018).

### **Solución nutritiva**

La solución nutritiva es uno de los elementos más importantes del sistema hidropónico, constituye la energía vital de las plantas que propician el crecimiento normal de cada célula. En la Tabla 1 se muestran los componentes minerales que debe contener una solución nutritiva: macroelementos como el potasio, calcio, magnesio, nitrógeno, fósforo, azufre; oligoelementos como el boro, cobre, zinc, molibdeno y microelementos como cloro, sodio y aluminio (Mohammed, 2018).

**Tabla 1**

*Funciones de los minerales en las plantas hidropónicas.*

<b>Mineral</b>	<b>Beneficio para la planta</b>
<b>Oxígeno</b>	Necesario para la producción de celulosa
<b>Carbono</b>	Vital para la constitución de células.
<b>Nitrógeno</b>	Presente en la clorofila y protoplasma, esencial para las proteínas.
<b>Fósforo</b>	Presente en semillas y frutos, primordial para todas las células vivas.
<b>Calcio</b>	Permite la construcción de paredes celulares y neutralización de ácidos orgánicos.
<b>Sulfuro</b>	Permite la formación de clorofila
<b>Oligoelementos</b>	Fundamentales porque actúan como agentes catalizadores.

*Nota.* La tabla exhibe los componentes fundamentales que se hallan presentes en una solución nutritiva para cultivos hidropónicos. Tomado de Benefits of minerals to plants. (p.3) por Mohammed. S, 2018, Springer.

Para facilitar la elaboración de soluciones nutritivas, la mayoría de ocasiones los componentes minerales vienen empaquetados en forma de sales fertilizantes como por ejemplo sulfato de amonio, cloruro de potasio, cloruro de magnesio, sulfato de hierro, quelato de magnesio, nitrato de potasio, sulfato de zinc, quelato de hierro. Todos los

nutrientes son absorbidos por medio de las raíces, las mismas que son sostenidas por medio de algún apoyo de tipo mecánico al sistema hidropónico. Según el tipo de cultivo hidropónico, costo, ubicación y diseño de sistema, se debe elegir el tipo de medio mecánico más adecuado, el mismo que debe tener un tamaño apropiado, limpio y no reactivo que permita la óptima absorción de oxígeno y los nutrientes (Resh, 2004).

Los mejores medios de soporte pueden ser realizados de sustancias de tipo inorgánico como grava de perlita o lana de roca o realizada de sustancias orgánicas como fibra de coco, musgo o corteza de pino (Alexander, 2000). Las características de estos materiales se presentan en la Tabla 2.

**Tabla 2**

*Características de medios de soporte en hidroponía*

<b>Medio</b>	<b>Característica</b>
<b>Lana de Roca “rockwool”</b>	Medio no degradable, requiere ser sumergido en agua con pH antes de ser pH balanceado.
<b>Perlita</b>	Al ser expuesta a altas temperaturas se torna muy ligera. Tiene excelentes niveles de pH, pero no una alta capacidad para retener el agua.
<b>Fibra de coco</b>	Material realizado en base a la cáscara de coco. Tiene alta capacidad para retener el agua y posibilita que el oxígeno circule libremente.
<b>Vermiculita</b>	Es un tipo de material de silicato que al ser expuesto al calor se expande. Capacidad media alta para retención de agua.
<b>Arena</b>	Material con partículas similares a las de una roca, pero más pequeña y con fácil drenaje.

*Nota.* La tabla expone las características sustanciales de los materiales para soporte de raíces en cultivos hidropónicos. Tomado de *Mediums used in hydroponics*. (p.3) por Mohammed. S, 2018, Springer.

### ***Smart Farming***

La sustitución de métodos agrícolas tradicionales con la inclusión de nuevas técnicas, conceptos, metodologías y tecnología innovadora para hacerla sostenible, se denomina agricultura de precisión o *Smart Farming*. Su enfoque busca maximizar los niveles de producción agrícola, reducir el uso de recursos y minimizar el impacto ambiental al limitar la irrigación y definir cantidades adecuadas de fertilizantes y pesticidas (Latief & Firasath, 2021).

Según Gandonou (Latief & Firasath, 2021), *Smart Farming* puede definirse como "conjunto de tecnologías que han impulsado la evolución de la agricultura convencional hacia el mundo de la transformación digital, diseñada para mejorar la gestión de diversas operaciones agrícolas".

Los componentes primordiales de *Smart Farming* son: generación de información, uso de tecnología y gestión de un sistema integral para el progreso de producción, calidad y eficiencia de los cultivos, disminuir el uso de energía y por ende la preservación del medio ambiente. La innovación de Smart Farming resulta de sumo beneficioso en países en desarrollo como Ecuador, ya que posibilita a los pequeños agricultores optimizar el rendimiento con un ínfimo empleo de insumos (Latief & Firasath, 2021).

De manera urgente se necesita realizar modificaciones en la manera de gestión agrícola con el objetivo de preservar de forma sostenible recursos no renovables (calidad del agua, aire y suelo). Es por ello que en *Smart Farming* aparece el concepto de las cinco "R": Aportación correcta (de productos fertilizantes y plaguicidas) *Right Input*, momento adecuado *Right time*, lugar adecuado *Right place*, cantidad correcta *Right amount* y manera correcta *Right manner* (Latief & Firasath, 2021).

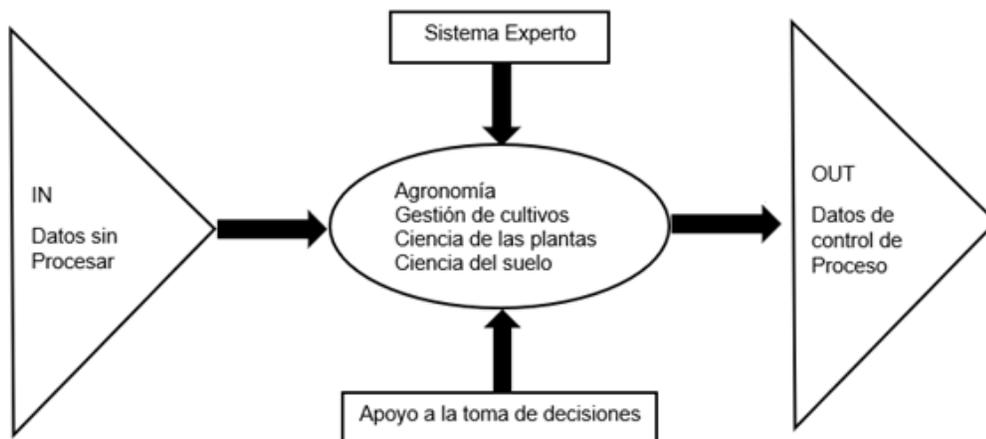
La agricultura inteligente se caracteriza por la recopilación de una gran cantidad de datos que provienen de diferentes dispositivos y sensores que se encargan de realizar mapeos de factores físicos del suelo, de los cultivos y del entorno del campo. Por ello, se enuncia a *Smart Farming* como "un sistema intenso de información" (Latief & Firasath, 2021). En la Figura 3 se realiza la representación del flujo de información.

Los procesos básicos y fuentes de información de la agricultura inteligente son:

1. Monitorización de parámetros y variables agrícolas e identificación de la ubicación de los sensores y recopilación de datos.
2. Transferencia de datos del campo de cultivo a la estación de control para la toma de decisiones.
3. Actuación y decisión controlada basada en los datos sensados.

### Figura 3

*Flujo de Información en Smart Farming*



*Nota.* La figura presenta el flujo de información proveniente de diferentes niveles de procesamiento y gestión que conforma el sistema *Smart Farming*. Tomado de Information Flow in Precision Agriculture (p.6) por A, Latief, 2021, CRC Press.

## LPWAN

Con la aparición de nuevas tecnologías como las comunicaciones máquina a máquina e IoT, está previsto un crecimiento masivo de redes LPWAN y el despliegue de millares de nodos sensores. Ericsson, en sus previsiones establece que para el año 2022 aproximadamente 29.000 millones de dispositivos estarán conectados a Internet, los dispositivos IoT interconectados serán coches, máquinas industriales, contadores, sensores, productos electrónicos de consumo y otros (Chaudhari & Zennaro, LPWAN Technologies for IoT and M2M Applications, 2020).

Para aplicaciones en los que se requiere detección y transmisión de datos en entornos de corto alcance se utilizan redes inalámbricas de área local (WLAN) IEEE 802.11, Bluetooth IEEE 802.15.1, ZigBee IEEE 802.15.3, redes de área personal inalámbricas de baja velocidad (*LR WPAN*, del inglés *Low Rate Wireless Personal Area Network*), entre otras. Por el contrario, para aplicaciones de largo alcance se utiliza tecnologías móviles celulares como 2G,3G,4G Y 5G. WLAN Y Bluetooth fueron diseñadas para comunicación de datos de alta velocidad, mientras que ZigBee y LR-WPAN diseñados para aplicaciones de detección inalámbrica en entornos locales con tasas de transmisión bajas. Estas tecnologías de comunicación dependen de las distancias con un alcance de hasta los cientos de metros, de la línea de vista, la presencia de obstáculos, ruido, interferencia y potencias de transmisión (Chaudhari & Zennaro, LPWAN Technologies for IoT and M2M Applications, 2020).

Cabe mencionar que para conseguir la comunicación de datos y voz se diseñaron las redes celulares, pero no para aplicaciones de detección con tecnología inalámbrica, debido a que su rendimiento en términos de métricas de desempeño con redes *WSN* (del inglés, *Wireless Sensor Networks*) no es óptimo. De manera que, para cumplir con estas métricas de desempeño, surge un nuevo paradigma de IoT

denominado LPWAN. Una LPWAN es una red de área amplia de baja potencia con estándares y soluciones de comunicaciones inalámbricas para IoT, con características fundamentales como: áreas de cobertura extensas, velocidades de transmisión bajas, transmisión de paquetes de datos de tamaño pequeño y larga duración de sus fuentes de energía. Con el despliegue de las tecnologías LPWAN se ha mostrado su enorme potencial para el desarrollo de proyectos de IoT y M2M, especialmente en entornos con comunicaciones limitadas (Chaudhari & Zennaro, LPWAN Technologies for IoT and M2M Applications, 2020).

### ***Aplicaciones y Servicios Inteligentes***

La creciente demanda de tecnología IoT es causada por su uso en dominios dependientes de una conectividad, que requieren grandes áreas de cobertura y capacidad para manejar un masivo número de conexiones. Con la evolución en el campo de la microelectrónica, las comunicaciones, la informática, desarrollo de nuevos sensores, actuadores y fuentes de energía es factible el diseño de redes con bajo consumo energético y con largo alcance lo que se traduce en decenas de kilómetros de cobertura y muchos años de duración de la batería. Estas redes presentan compatibilidad con protocolos de Internet de tal forma que la gestión de datos, dispositivos y la red pueda efectuarse con el uso de plataformas basadas en la nube (Chaudhari & Zennaro, LPWAN Technologies for IoT and M2M Applications, 2020).

Los requerimientos imprescindibles de los dispositivos inalámbricos IoT/M2M son el bajo consumo de energía, mayor alcance de transmisión, compatibilidad con otros dispositivos de la red, capacidad para manejar las interferencias de RF, bajo coste, facilidad de despliegue y alta seguridad informática. LPWAN es una tecnología innovadora que puede implementarse en una amplia gama de aplicaciones inteligentes desde la monitorización de parámetros ambientales, ciudades inteligentes, servicios

públicos, agricultura, atención médica, automatización industrial, transporte y muchas otras aplicaciones como se presenta en la Tabla 3.

**Tabla 3**

*Aplicaciones de LPWAN*

<b>Campo</b>	<b>Principales Aplicaciones</b>
<b>Ciudades inteligentes</b>	Aparcamientos inteligentes, salud estructural de edificios, puentes y monumentos históricos, medición de calidad del aire, medición del nivel de ruido sonoro, control de semáforos y del tráfico.
<b>Agricultura y ganadería inteligentes</b>	Temperatura, humedad, alcalinidad, invernaderos inteligentes, automatización y robótica agrícola, estaciones meteorológicas, compost, hidroponía, cuidado de crías, control y seguimiento del ganado.
<b>Casas/edificios e inmuebles Inteligentes</b>	Control del uso de energía y agua, temperatura, humedad, detección de incendios y humo, control remoto de dispositivos, sistemas de detección, preservación de bienes.
<b>Salud, Ciencias de la Vida y wearables</b>	Salud y parámetros médicos de pacientes, wearables para la atención sanitaria, vigilancia de pacientes, monitorización de la radiación ultravioleta.
<b>Automóviles y Logística</b>	Gestión de vehículos, calidad de las condiciones de envío, localización de artículos, seguimiento de flotas, buses inteligentes
<b>Seguridad y Emergencias</b>	Controles de acceso, detección de presencia de líquidos, control del nivel de radiación y gases peligrosos.

*Nota.* La tabla presenta los campos de aplicación en el mundo físico con redes LPWAN. Tomado de *LPWAN Technologies for IoT and M2M Applications*. (p.25) por Bharat .C & Zennaro .M, 2020, Academic Press.

## Capítulo III

### Materiales y Métodos

#### Comparación de tecnologías LPWAN

Uno de los componentes principales para el desarrollo del sistema de sensores inalámbricos es la tecnología LPWAN. En el mercado existen diferentes soluciones LPWAN con distintas características respecto a costos de los dispositivos, velocidades de transmisión, duración de la batería, alcance, etc. Las principales soluciones se clasifican en 2 categorías: propietarias y basadas en estándares. Las tecnologías LPWAN propietarias son: Sigfox, Ingenu, Telensa, Qowisio y Nwave, mientras que las basadas en estándares son: LoRa, *Weightless*, *Narrowband Internet of Things* (NB-IoT), LTE-M, DASH7. En la Tabla 4 se expone una comparativa de las soluciones con mayor aplicación en redes IoT basado en el análisis realizado por (Maclay, 2017) en el cual se realiza una comparación minuciosa de diferentes aspectos técnicos importantes a ser considerados antes de realizar un sistema con cualquiera de estas tecnologías.

**Tabla 4**

*Comparativa de redes LPWAN*

	<b>Sigfox</b>	<b>LoRa</b>	<b>NB-IoT</b>	<b>Weightless</b>	<b>Telensa</b>
<b>Rango de Cobertura</b>	Zona Urbana (10km) Zona Rural (40km)	Zona Urbana (3km) Zona Rural (50km)	Zona Urbana (2km) Zona Rural (50km)	Zona Urbana (2km) Zona Rural (3km)	Zona Urbana (2km) Zona Rural (4km)
<b>Data Rate</b>	300bps	200bps-50kbps	20-250kbps	10Mbps	80kbps
<b>Tipo de Red</b>	Pública	Pública o Privada	Pública	Pública	Pública
<b>Disponibilidad</b>	Cobertura limitada	Pública limitada	Cobertura limitada	Cobertura limitada	Cobertura limitada
<b>Admite Audio</b>	No	No	Si	Si	No
<b>Modulación</b>	DBPSK	CSS	QPSK	DBPSK	QPSK

	<b>Sigfox</b>	<b>LoRa</b>	<b>NB-IoT</b>	<b>Weightless</b>	<b>Telensa</b>
<b>Frecuencia de Operación</b>	Bandas ISM sin licencia	Bandas ISM sin licencia	Bandas ISM sin licencia	Espacio blanco de televisión (TVWS)	Bandas ISM sin licencia
<b>Comunicación bidireccional</b>	Limitada de tipo Halfduplex	Halfduplex	Halfduplex	Unidireccional	Halfduplex
<b>Tasa de mensajes</b>	140 por día en UL y 4 DL	Ilimitada	Ilimitada	Ilimitada	100 por día en UL y 4 DL
<b>Tamaño de la carga útil en bytes</b>	En UL de 12 y en DL de 8	243	1600	12	8
<b>Seguridad</b>	No soporta autenticación ni encriptación	Soporta autenticación y encriptación	Encriptación de LTE	Encriptación	Encriptación
<b>Interferencia</b>	Muy alta inmunidad	Muy alta inmunidad	Baja inmunidad	Baja inmunidad	Alta inmunidad

*Nota.* La tabla muestra una comparativa entre aspectos técnicos importantes de las soluciones LPWAN existentes en el mercado de telecomunicaciones. Tomado de *Comparison of IoT Wireless Standards*. (p.10) por Maclay. W, 2017, Voler Systems.

También es importante tener presente la inversión económica de los equipos e infraestructura para el despliegue con las tecnologías LPWAN anteriormente analizadas, en la Tabla 5 se realiza una comparativa de costos entre las principales soluciones de área amplia de baja potencia.

### **Tabla 5**

*Comparación de Costos entre tecnologías LPWAN*

Parámetro	Sigfox	LoRa	Nb-IoT	Weightless
<b>Despliegue</b>	Mayor a 4000\$ por estación base	Gateway: mayor a 100 dólares	Mayor a 15000\$ por estación base	Mayor a 500\$
<b>Espectro</b>	Gratuito	Gratuito	Mayor a 500M\$ por MHz	Gratuito
<b>Módulos</b>	<40\$	<40\$	>30\$	>30\$
<b>Conectividad</b>	<1\$/mes	\$0.25/mes	<1\$/mes por 100kb	\$0.25/mes

*Nota.* La tabla muestra una comparativa entre aspectos técnicos importantes de las soluciones LPWAN existentes en el mercado de telecomunicaciones. Tomado de *Comparison of IoT Wireless Standards*. (p.10) por Maclay. W, 2017, Voler Systems.

En base a las comparativas técnicas-económicas analizadas anteriormente, se elige como tecnología LPWAN a la red LoRa por sus ventajas para el despliegue, bajo costo de dispositivos, capacidad para realizar un mantenimiento de red autónomo y bidireccionalidad para escenarios de mando y control. La red de sensores inalámbricos del proyecto es implementada en interiores, por lo cual la conectividad de LoRa no se ve afectada por la interferencia. Además, los dispositivos de LoRa presentan un menor consumo de energía con respecto a dispositivos NB-IoT y Sigfox, lo que aumenta la duración de la batería. La seguridad es otro parámetro técnico esencial, LoRa dispone de encriptación y autenticación para identificar los nodos finales de la red y con capacidad de cifrado de los datos que se transmiten al servidor. El envío de datos hacia Internet provenientes de los sensores de monitorización debe realizarse para el caso de *Smart Farming* por lo menos cada 5 minutos, por lo que se requiere de un sistema con

una tasa de mensajes ilimitada, siendo LoRa la solución que se ajusta a los requerimientos del proyecto.

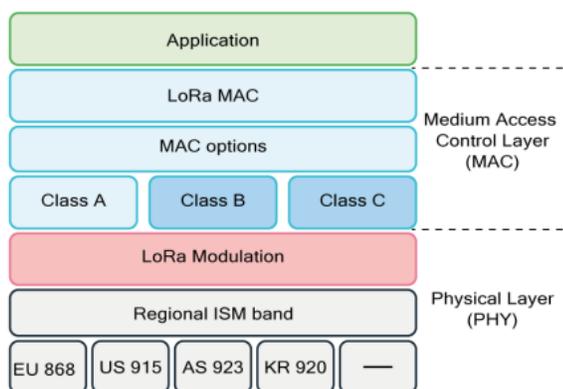
## Análisis de la tecnología LPWAN LoRa

### LoRa

LoRa representa la capa física de una red LoRaWAN creada por la empresa tecnológica Semtech y que utiliza la modulación *Chirp Spread Spectrum (CSS)*. La Figura 4, presenta la pila de protocolos de LoRa, constituida de la capa de aplicación, la capa MAC, la capa *PHY* y la capa *RF*.

### Figura 4

*Pila de Protocolos LoRa/LoRaWAN*



*Nota.* La pila de protocolos LoRaWAN (MAC) es implementada sobre la capa física. Tomado de LoRa/LoRaWAN protocol stack (p.323) por P, Lea, 2020, Packt.

La capa física se encarga de procesos de gestión de la modulación, potencia, radios de recepción y transmisión y el acondicionamiento de la señal. *LoRa* presenta un ancho de banda de tipo escalable, tasa de datos variable y un factor de dispersión también conocido como factor ortogonal de ensanchamiento (Lea, 2020)

Con el propósito de optimizar la utilización del espectro y rendimiento de la red, LoRa posibilita seleccionar un total de seis factores de dispersión (SF, del inglés *Spreading Factor*) de 7 a 12 que determina el número de bits que se necesitan para transmitir los datos. El factor de dispersión es la relación que existe entre la tasa de chips “ $R_C$ ” y la tasa de símbolos “ $R_S$ ”:

$$SF = \log_2\left(\frac{R_C}{R_S}\right) \quad (1)$$

El *bitrate* utilizado por LoRa está en función de la tasa de chirp y de la tasa de símbolos. La tasa de bits se denota con  $R_b$ , el factor de dispersión con  $SF$  y el ancho de banda con  $BW$ . Por lo tanto, la tasa de bits (bps) puede variar entre los 0.3kbps hasta los 5kbps y se obtiene mediante la ecuación (2):

$$R_b = SF \times \frac{1}{\left[\frac{2^{SF}}{BW}\right]} \quad (2)$$

Según (Chaudhari & Zennaro, 2020) “el transmisor genera señales chirp al variar su frecuencia en el tiempo y mantener constante la fase entre símbolos adyacentes”. En la Tabla 6, se plantea la relación existente entre la tasa de bits, el factor de dispersión y la potencia.

**Tabla 6.**

*Relación entre Factor de Dispersión, Tasa de Bits, Sensibilidad y SNR*

Factor de Dispersión “SF”	Date Rate [Kbps]	Sensibilidad [dBm]	SNR [dBm]
7	5.468	-123	-7.5
8	3.125	-126	-10
9	1.757	-129	-12.5

<b>Factor de Dispersión "SF"</b>	<b>Date Rate [Kbps]</b>	<b>Sensibilidad [dBm]</b>	<b>SNR [dBm]</b>
<b>10</b>	0.976	-132	-15
<b>11</b>	0.537	-134.5	-17.5
<b>12</b>	0.293	-137	-20

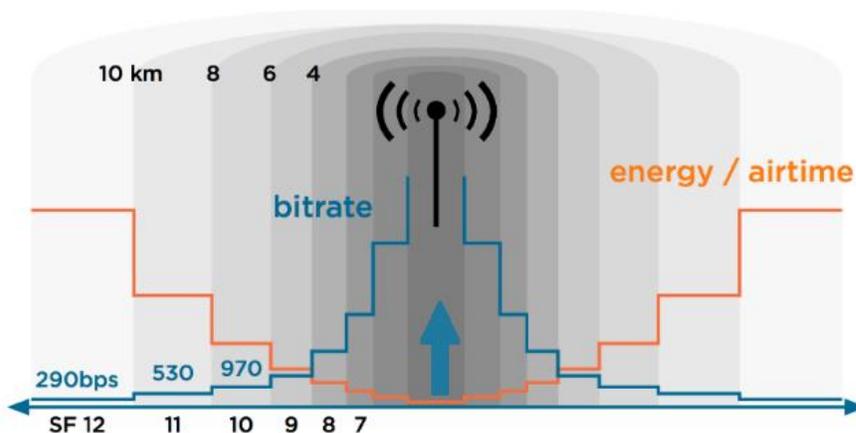
*Nota.* La tabla presenta el máximo y mínimo valor del factor de dispersión, cada variación en una unidad involucra la reducción de 3dBs en la sensibilidad del receptor. Tomado de Relación entre SF y sensibilidad, por Moya. M (2018, p.9).

La capacidad del receptor para demodular el mensaje aumenta cuando existe un mayor número de bits por símbolo y puede decodificar la señal incluso si presenta atenuaciones de 19.5 dB por debajo del nivel de ruido. Las tramas en LoRa son ortogonales, lo que significa que existe la posibilidad de enviar diversas tramas de forma simultánea siempre que cada una sea enviada usando un factor de propagación diferente (Lea, 2020).

Cuando existe una gran distancia entre el *Gateway* y los nodos LoRa se usa SF superiores lo que resulta en una menor velocidad de datos, aumento del tiempo en el aire y se logra que la señal presente mayor solidez con respecto a las interferencias, mientras que los nodos LoRa con mayor cercanía al *Gateway* usan SF más bajos, de modo que es posible aumentar la tasa de datos, se disminuye el tiempo en el aire y se optimiza el consumo energético, esto se aprecia en la Figura 5.

**Figura 5**

Relación SF, bitrate y energía



*Nota.* Con valores de SF altos, la transmisión es más lenta debido a la necesidad de mayor cantidad de bits para enviar la información, pero se obtiene un mayor alcance. Tomado de LoRa Spreading Factors, por Moya. M (2018, p.10).

Los tres anchos de banda disponibles son de 125, 250 y 500 kHz, pero por lo general, en las redes LoRa se hace uso de 125 kHz como ancho de banda. La relación que existe entre SF, BW y el tamaño del paquete de datos permite precisar el tiempo en el aire (*ToA*, del inglés *Time on Air*). *ToA*, es definido como el tiempo total de uso de la interfaz aérea para hacer el envío de un paquete en una sola transmisión. Al utilizar un mismo ancho de banda para transmitir dos paquetes del mismo tamaño, pero con SF el *ToA* es totalmente diferente (Rodríguez & Kleinschmidt, 2020). El rendimiento del enlace depende del *BW*, tamaño del paquete de datos y el SF escogidos, una comparación detallada se realiza en la Tabla 7.

**Tabla 7***Relación entre SF, Throughput y ToA*

SF	Ancho de Banda [kHz]	Tamaño máximo de MAC Payload [bytes]	Tamaño máximo de FRM Payload [bytes]	Máximo ToA downlink [s]	Máximo ToA uplink [s]	Throughput [bps]
7	250	250	242	0.197	0.200	5468
7	125	250	242	0.394	0.400	4913.7
8	125	250	242	0.697	0.707	2777.6
9	125	123	115	0.677	0.677	1358.9
10	125	59	51	0.698	0.698	584.5
11	125	59	51	1.479	1.561	275.9
12	125	59	51	2.793	2.793	146.1

*Nota.* Para la transmisión de un paquete de datos con un factor de dispersión de 12 la latencia es 20 veces mayor que con un SF de 7. Tomado de LoRaWAN data rates settings and frames characteristics. Por Mikhaylov. Konstantin, Petäjäjärvi. Juha & Hänninen. Tuomo. J (2016, p.3).

En LoRa, se emplea la técnica de corrección de errores hacia adelante (FEC, del inglés, *Forward Error Correction*), con la finalidad de aumentar la sensibilidad en el receptor. La cantidad de FEC en la trama LoRa viene definido por la tasa de código (CR, del inglés *Code Rate*). LoRa dispone de CR igual a 0,1,2,3 y 4, donde un CR igual a 0 significa que no existe codificación. ToA, asimismo, depende de SF y CR, en base a estos parámetros la tasa de bits  $R_b$  también puede calcularse por medio de la ecuación (3):

$$R_b = \frac{4 \times SF \times BW}{(4 + CR) \times 2^{SF}} (bps) \quad (3)$$

### Radiofrecuencia

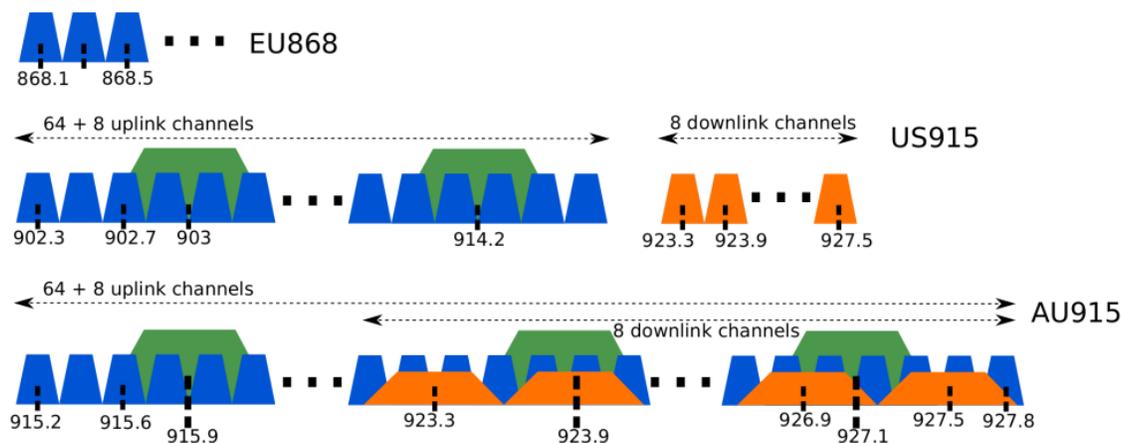
LoRa usa frecuencias industriales, científicas y médicas (ISM), la banda de frecuencias depende de cada país o región. La arquitectura de LoRa se encuentra basada en las siguientes bandas del espectro libre de licencias ISM: (Lea, 2020)

- 915 MHz: en Estados Unidos y América del Sur con límites en cuanto a potencia, pero sin ningún límite en el ciclo de trabajo.
- 868 MHz: En Europa con límites en el ciclo de trabajo de un 1% a 10%
- 433 MHz: En Asia

En cada país existe un esquema de subbandas de frecuencias para formar canales de transmisión. Cada subbanda está compuesta de cierto número de frecuencias llamadas canales. En la Figura 6, se expone el esquema de canales para Europa (EU868), Estados Unidos (US915) y Australia (AU915) (LoRa, 2020)

**Figura 6**

*Canales para Europa, Estados Unidos y Australia.*



*Nota.* En Europa, la normativa establece que los canales de red pueden ser atribuidos libremente, sin embargo, los nodos finales deben implementar los tres canales por defecto (868.1, 868.3 y 868.5 MHz). Tomado de Long-Range Wide Area Network (LoRaWAN) channels for Europe, USA, and Australia, por Rodriguez. J & Kleinschmidt. J (2020, p.3).

En Estados Unidos y Australia se tiene un esquema de 64 canales para el enlace ascendente, con un ancho de banda de 125 kHz y que se incrementa linealmente cada 200 kHz. En ambas regiones, existen ocho canales adicionales para el enlace ascendente de 500 kHz cada uno y que se incrementan cada 1.6 MHz. Para la comunicación entre el *Gateway* y los nodos finales, existen ocho canales para el enlace descendente de 500 kHz, desde los 923.3 MHz los cuales se incrementan cada 600 kHz. La Tabla 8, resume los canales, las frecuencias y las subbandas de LoRa para la frecuencia AU 915-928 MHz.

**Tabla 8**

*Canales, frecuencias y subbandas de LoRa*

<b>Bandas de frecuencia</b>	<b>Rango de frecuencia (MHz)</b>	<b>Canales</b>
Banda total US	915.2 - 928	0-63
<b>Subbanda de Uplink</b>	<b>Rango de frecuencia (MHz)</b>	<b>Canales de Uplink de 125 kHz</b>
Sub-Banda 1	915.2 - 916.6	0-7
Sub-Banda 2	916.8 - 918.2	8-15
Sub-Banda 3	918.4 - 919.8	16-23
Sub-Banda 4	920.0 - 921.4	24-31
Sub-Banda 5	921.6 - 923.0	32-39
Sub-Banda 6	923.2 - 924.6	40-47
Sub-Banda 7	924.8 - 926.2	48-55

Sub-Banda 8	926.4 - 927.8	56-63
<b>Subbanda de Uplink</b>	<b>Rango de frecuencia (MHz)</b>	<b>Canales de Uplink de 500 kHz</b>
-	915.9 - 927.1	64-71
<b>Subbanda de Downlink</b>	<b>Rango de frecuencia (MHz)</b>	<b>Canales de Downlink de 500 kHz</b>
-	923.3 - 927.5	0-7

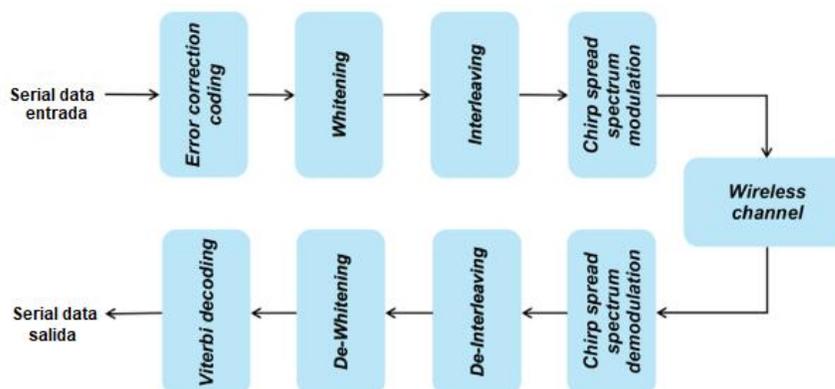
*Nota.* Tomado de *LoRaWAN Channels, Frequencies & Sub-bands*. Por Barani. J, (2019).

### Capa física (PHY) de LoRa

La tecnología LoRa es propietaria de Semtech, creada para permitir la conexión de dispositivos a la nube y la transmisión en tiempo real de datos. En la Figura 7, se presenta el diagrama de bloques de la arquitectura de un transceptor LoRa.

### Figura 7

*Arquitectura de la capa PHY de LoRa.*



*Nota.* Diagrama de bloques de la capa física de LoRa. Tomado de *LoRa PHY layer architecture*. (p.84) por Chaudhari. B & Zennaro. M, Academic Press.

## Encoding

Los bits de entrada provenientes de la fuente binaria ingresan a un codificador. El proceso de codificación reduce la tasa de error en los paquetes de datos cuando se produce ráfagas cortas de interferencia. La salida del codificador está sujeta al valor de CR elegido. LoRa, para FEC emplea códigos Hamming de bloque lineales y de implementación sencilla. Las tasas de codificación (*coding rates*) utilizadas por LoRa son de 4/7, 1/2, 2/3 y 4/5. La tasa de código es denotada con  $k/n$ , donde  $k$  define el número de bits de datos útil y  $n$  son los bits de salida generados por el codificador, por consiguiente, los bits redundantes son  $(n-k)$ . La Tabla 9, expone las capacidades para detectar y corregir errores cuando se tiene valores de CR entre 1,2,3 y 4 y tasas de codificación 4/5, 4/6, 4/7 y 4/8 respectivamente (Chaudhari & Zennaro, LPWAN Technologies for IoT and M2M Applications, 2020).

**Tabla 9**

*Capacidades de detección y corrección de errores de LoRa*

<b>Coding rates</b>	<b>Detección de errores (bits)</b>	<b>Corrección de errores (bits)</b>
<b>4/5</b>	0	0
<b>4/6</b>	1	0
<b>4/7</b>	2	1
<b>4/8</b>	3	1

*Nota.* Tomado de *Error detection and correction capabilities of LoRa*. (p.85) por Chaudhari. B & Zennaro. M, Academic Press.

## Whitening

Los bits de salida del codificador luego ingresan al bloque opcional de *whitening* "blanqueo" en LoRa, la implementación de este bloque se lo realiza con la codificación

Manchester para originar aleatoriedad, puesto que, el objetivo del whitening es asegurar que no exista largas cadenas de 0s o 1s en el payload (carga útil).

### **Interleaving**

La salida del bloque de *Whitening* luego ingresa al bloque de Interleaving. El intercalador (interleaver) emplea el método diagonal placing para realizar el cambio de secuencia de ordenación de los bits codificados (*scramble*) cada  $4+CR$  palabras de código. A fin de que la ráfaga de errores se distribuya entre varias palabras de código en lugar de limitarse a sólo una.

### **Modulación Chirp Spread Spectrum**

LoRa se define como un esquema de modulación de espectro ensanchado basado en la modulación CSS. CSS fue desarrollada para ser utilizada en aplicaciones de radar en los años 40. Por sus bajos requerimientos de potencia de transmisión, poca interferencia y capacidad para largas distancias de comunicación ha sido utilizada por décadas en comunicaciones militares y espaciales.

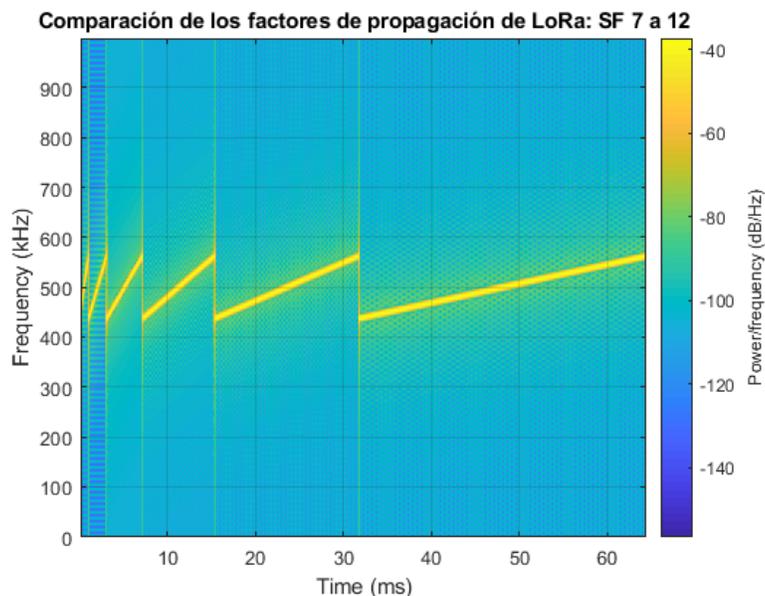
CSS es una modulación de espectro ensanchado chirp que conserva las características propias de bajo consumo que la modulación FSK (del inglés *Frequency Shift Keying*), emplea pulsos chirp de tipo lineal de banda ancha que son modulados en frecuencia para realizar el proceso de codificación de la información. CSS hace uso de factores de dispersión que son ortogonales entre sí para difundir la información sobre la frecuencia, como se presenta en la Figura 8 (Kim, Lee, & Kim, 2019).

Un pulso de radar de alta intensidad comprimido (*CHIRP*, del inglés *Compressed High Intensity Radar Pulse*), frecuentemente llamado señal de barrido (*sweep signal*), constituye un tono o pulso en el que la frecuencia aumenta (*upchirp*) o disminuye

(*downchirp*) en el tiempo. La modulación de LoRa emplea la técnica de espectro de dispersión *chirp* para realizar la codificación de los datos (Ghoslya, 2017).

### Figura 8

*Espectrograma de los factores de propagación de LoRa y CSS*



*Nota.* Los símbolos LoRa son modulados sobre un chirp ascendente "up-chirp" de 125 kHz de ancho de banda, donde se observa que con un SF igual a 9 existe una duración del doble de tiempo que con un SF igual a 8. Simulación realizada basada en el código *Matlab Code to Generate LoRa Symbols* por Ghoslya. S (2019).

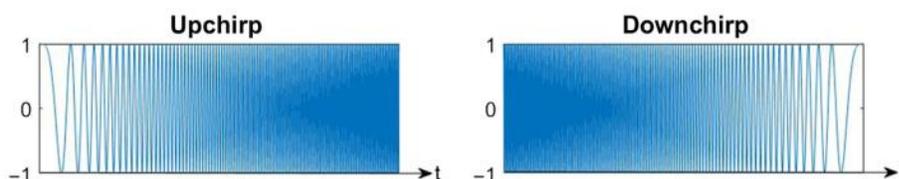
Las señales chirp se caracterizan por tener amplitud constante y recorrer todo el ancho de banda de manera lineal o no lineal de un extremo a otro en tiempo determinado. Chirp es una señal de tipo sinusoidal con una frecuencia lineal que puede ser creciente o decreciente expresada mediante la ecuación (4).

$$c(t) = \begin{cases} \exp(2\pi j(at + b)t), & -\frac{T_s}{2} \leq t \leq \frac{T_s}{2} \\ 0, & \text{Otro caso} \end{cases} \quad (4)$$

Con  $at + b = f_{min} + \frac{f_{max} - f_{min}}{T_s} t$ , donde  $f_{max}$  y  $f_{min}$  son la máxima (125 kHz para el proyecto) y mínima frecuencia respectivamente.  $T_s$  representa la duración de símbolo. El espectro ensanchado *chirp* emplea todo el ancho de banda para transmitir las señales. Cuando la frecuencia cambia de la más baja a la más alta, se denomina *upchirp* y cuando cambia de la más alta a la más baja se denomina *downchirp*. En la Figura 9, se observa las señales de tipo *Upchirp* y *Downchirp* lineales.

### Figura 9

Señal *Upchirp* y *Downchirp* lineal

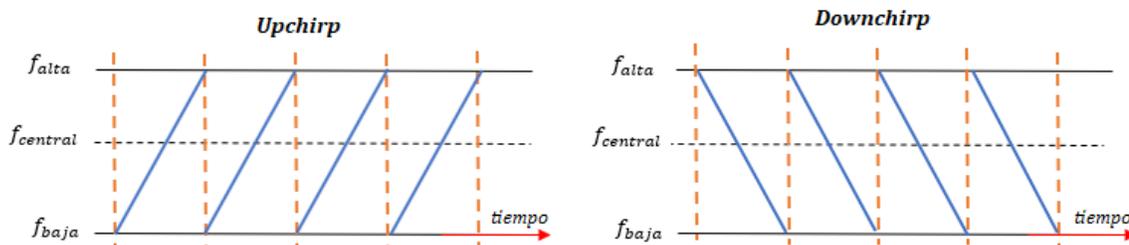


*Nota.* Tomado de *Upchirp and downchirp signal*. (p.85) por Chaudhari. B & Zennaro. M, Academic Press.

En la Figura 10, se visualiza que existe una frecuencia máxima y mínima, en *Upchirp* la frecuencia aumenta en el tiempo y cuando alcanza la frecuencia máxima salta a la frecuencia más baja y el proceso se repite. En *Downchirp* la frecuencia disminuye en el tiempo, cuando alcanza la frecuencia mínima salta de nuevo a la frecuencia máxima y se repite el proceso.

**Figura 10**

Señales Upchirp y Downchirp en frecuencia

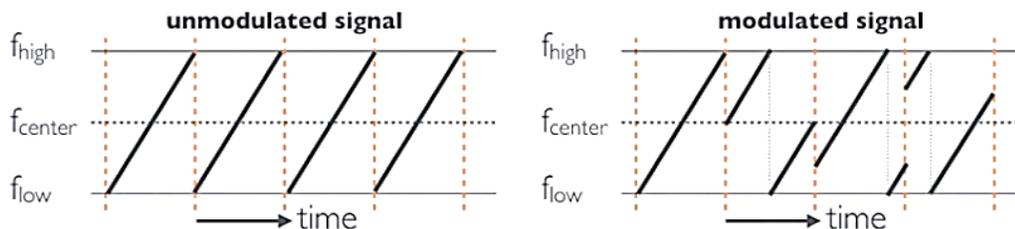


*Nota.* Tomado de *Upchirp and Downchirp signal.* (p.86) por Chaudhari. B & Zennaro. M, Academic Press.

La señal *Chirp* se utiliza como señal portadora en la que se realiza la codificación del mensaje. Los *chirps* se desplazan de manera cíclica, en la Figura 11, se observa como la modulación de los datos en los chirps depende de los saltos de frecuencia.

**Figura 11**

Modulación de la señal



*Nota.* Un nodo final en cada transmisión cambia de canal de forma pseudoaleatoria, con el salto o cambio de frecuencias el sistema se vuelve más resistente a las interferencias. Tomado de *Message encoded on the chirp signals.* (p.86) por Chaudhari. B & Zennaro. M, Academic Press.

Un símbolo representa a uno o más bits de datos, por ejemplo, un símbolo podría codificar la cadena de bits 1011111 con un equivalente decimal a 95, es decir 7 bits. Cada símbolo se distribuye en un *Upchirp* y su valor depende del valor de SF utilizado. El símbolo tiene  $2^{SF}$  chips y es codificado en la señal sweep (*Upchirp*), así con un factor SF igual a 7 se tiene 128 chips/símbolo, mientras que con un SF igual a 12 se puede representar 4096 chips/símbolo, respectivamente.

Existe una relación entre la tasa de bits de datos  $R_b$ , la tasa de símbolos  $R_s$  y la tasa de chirps  $R_c$ , expresado en la ecuación (2). La tasa de símbolos  $R_s$  viene definido por la ecuación (5).

$$R_s = \frac{1}{T_s} = \frac{BW}{2^{SF}} \frac{\text{símbolos}}{s}, \quad (5)$$

donde  $T_s$  corresponde al periodo de símbolo en segundos. La tasa de chirp  $R_c$ , viene definida por la ecuación (6).

$$R_c = R_s \times 2^{SF} \text{ Chirp/s} \quad (6)$$

En base a las ecuaciones anteriormente planteadas, SF y BW constituyen los parámetros principales en la modulación LoRa. Al reemplazar la ecuación (5) en la ecuación (6), se puede concluir que la tasa de chirp es igual al ancho de banda de la señal, ecuación (7).

$$R_c = \frac{BW}{2^{SF}} \times 2^{SF} = BW \quad (7)$$

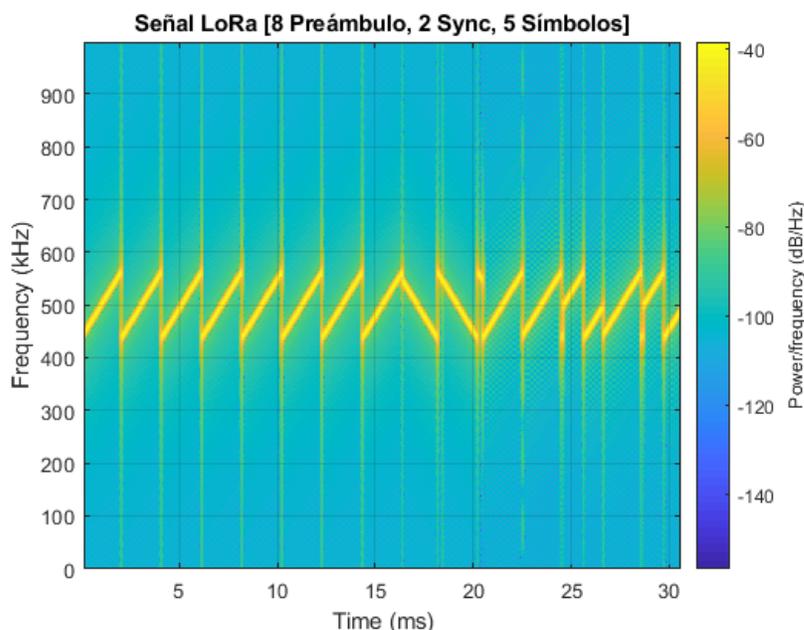
### Trama Física

La trama de una señal LoRa está constituida de 3 partes de una señal CSS, Figura 12. La primera parte constituye al preámbulo, la segunda la sincronización y la tercera son los datos. El preámbulo se encuentra constituido de 8 símbolos *Upchirp* y

son utilizados para establecer cuando inicia la señal, luego se tiene dos símbolos *Downchirp* para sincronizar la señal con el demodulador y los datos sean reconocidos de manera correcta. Los datos están compuestos de símbolos *chirp* discontinuos formados por los saltos de frecuencia y que contienen la información (Quinones, 2019).

## Figura 12

Señal LoRa



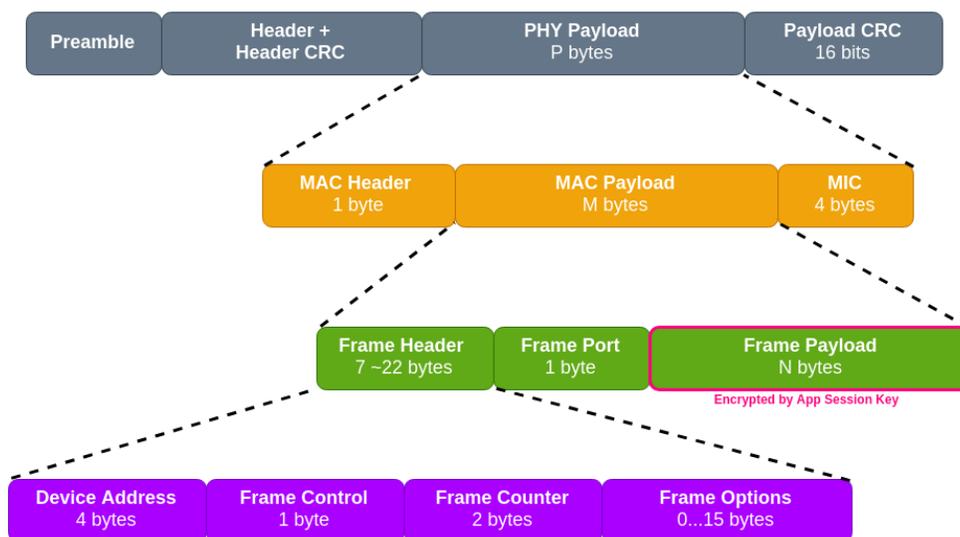
*Nota.* Los símbolos LoRa son modulados sobre un chirp ascendente "up-chirp" de 125 kHz de ancho de banda. Simulación realizada basado en el código *Matlab Code to Generate LoRa Symbols* por Ghosly. S (2019).

Los mensajes de enlace ascendente son transmitidos desde los nodos finales hacia el servidor de la red por medio de una o varios *Gateways* o pasarelas, mientras que los mensajes de *downlink* son transmitidos desde el servidor hacia un nodo final. En la Figura 13, se observa la estructura de la trama física para mensajes del enlace ascendente, el cual está constituido del preámbulo, la cabecera (*Header*) con

información acerca de los bytes, el código de redundancia cíclica (*Header CRC*), el campo de carga útil (*PHY Payload*) y códigos para detectar errores (*Payload CRC*) (Fernández & Uquillas, 2020).

### Figura 13

Formato de la Trama Física de LoRa

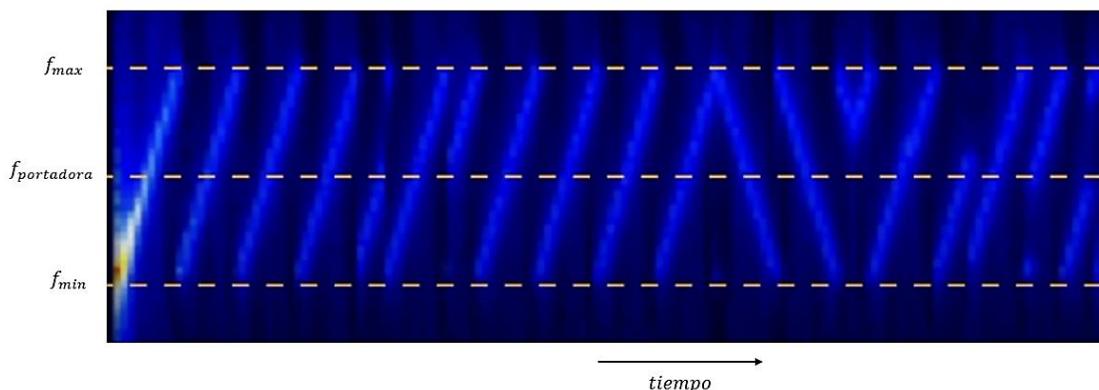


*Nota.* Tomado de LoRa *Frame Format* por LoRa Alliance (2018).

Por el contrario, en los mensajes de *Downlink* el formato de la trama física es similar al de *Uplink*, excepto por la inexistencia del campo de códigos para detección de errores (*Payload CRC*) (Fernández & Uquillas, 2020). En la Figura 14, se presenta un mensaje de *Uplink* codificado en las señales *Chirp*.

## Figura 14

*Mensaje codificado en las señales de chirp*



*Nota.* Tomado de *Message encoded on the chirp signals*. (p.86) por Chaudhari. B & Zennaro. M, Academic Press.

## Estándares

### LoRaWAN

LoRaWAN es un estándar global enfocado en el desarrollo de redes IoT de baja potencia y de área amplia LPWAN. En el mercado de telecomunicaciones IoT coexisten diversas tecnologías, estándares y protocolos de comunicación para las múltiples soluciones en las que se exigen sensores y actuadores con bajo consumo.

Según (Seneviratne, 2019) LoRaWAN la red de área amplia de largo alcance, “constituye el protocolo de comunicación y arquitectura del sistema para la red”. De manera general, la capa física de LoRa posibilita los enlaces de comunicación de largo alcance para la transmisión de comunicación, mientras que LoRaWAN tiene incidencia en los siguientes parámetros:

- Duración de la batería del nodo
- Capacidad de la red
- QoS, calidad del servicio

- Niveles de Seguridad
- Aplicaciones atendidas por la red

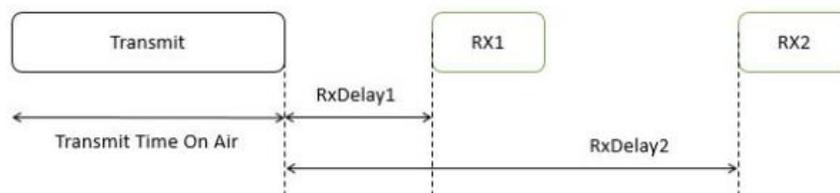
LoRaWAN se encuentra ubicada sobre la PHY LoRa, es un protocolo abierto a diferencia de PHY que es cerrado. Constituye el estándar de capa MAC que realiza la coordinación del acceso al medio, el cual opera con tres diferentes clases cuyo uso está sujeto al modo de comunicación que se quiera realizar entre la estación base y los nodos finales. Las clases conforman la capa de enlace de datos y realizan el equilibrio entre latencia y el consumo de energía.

Para la mitigación de la energía la clase A es la más apropiada, aunque presenta una latencia más alta, por el contrario, la clase C presenta una latencia mínima pero un mayor consumo de energía. La clase B es un equilibrio entre la clase A y C (Lea, 2020).

**Clase A:** El modo de alimentación por batería, es el más empleado en redes LoRaWAN. Consiste en el proceso de comunicación programado por medio del protocolo MAC denominado como ALOHA, que usa ACK para confirmar que se realiza la transmisión de datos. La comunicación funciona en el canal de *Uplink* por el nodo final, seguido de dos lapsos (*laps*) de tiempo de *Downlink* transmitido por la estación base. Los dispositivos que pertenecen a la clase A son sensores y nodos finales que hacen uso de baterías. Los nodos finales que se enlazan a la red LoRaWAN se asocian primero como Clase A con la alternativa de cambiar de clase a lo largo del funcionamiento, en la Figura 15 se presenta el proceso de comunicación en una clase A.

**Figura 15**

LoRaWAN Clase A

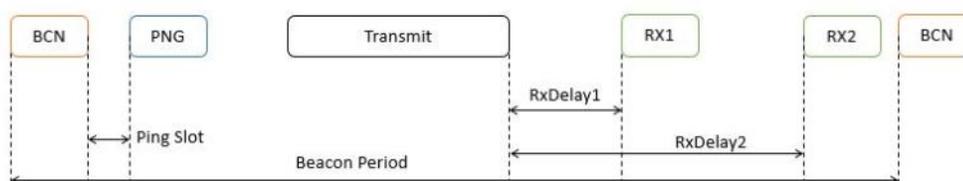


Nota. Tomado de LoRaWAN Class A. Por Quinones. M (2019, p.25)

**Clase B:** modo de baja latencia, presenta un funcionamiento semejante a la Clase A, pero la diferencia radica en un tiempo adicional para *Downlink* por parte del nodo final. Los dispositivos de la clase B efectúan un envío de tramas beacon por parte del *Gateway* en intervalos regulares. La trama beacon realiza la sincronización de todos los nodos finales de la red, en la Figura 16 se presenta el proceso de comunicación en una clase B.

**Figura 16**

LoRaWAN Clase B



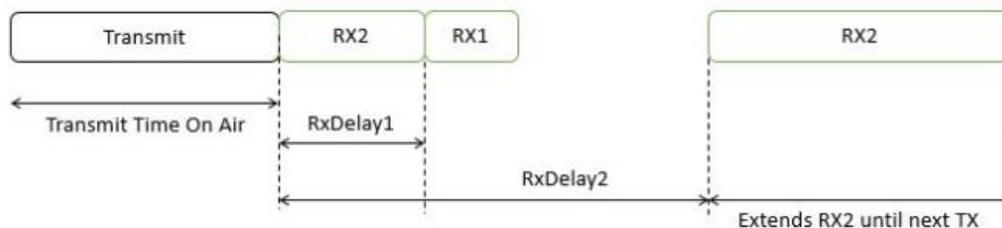
Nota. Tomado de LoRaWAN Class B. Por Quinones. M (2019, p.25)

**Clase C:** Modo de No latencia, posibilita la existencia de un tiempo continuo de *Downlink* en el nodo final lo que permite realizar la escucha permanente por el *Gateway*. Los dispositivos finales de la clase C son los que más energía consumen, presentan menor latencia, suelen hallarse encendidos y pueden ser actuadores o dispositivos

enchufables. No existe latencia cuando se realiza transmisión del enlace de *Downlink*, en la Figura 17 se presenta el proceso de comunicación en una Clase C.

**Figura 17**

*LoRaWAN Clase C*



*Nota.* Tomado de *LoRaWAN Class C*. Por Quinones. M (2019, p.25)

### Estructura de la red LoRaWAN

Una red LoRaWAN está conformada por elementos principales como nodos finales, *Gateways* (pasarelas), servidor de red y servidores de aplicaciones como se presenta en la Figura 18. La gran cantidad de datos transmitidos por un nodo final son recibidos por el *Gateway* o pasarela. Los datos recibidos por cada *Gateway* son empaquetados y reenviados al servidor de la red a través de una red celular, vía Ethernet, WiFi o satélites. La configuración de software en el *Gateway* posibilita el reenvío de cualquier paquete de datos recibidos al servidor de red. Este software se denomina como reenviador de paquetes (*packet forwarder*) (Seneviratne, 2019).

El *packet forwarder* constituye un software que se encuentra ejecutándose en el *Gateway* de LoRa, posibilita que la pasarela LoRa transmita y reciba paquetes de datos tanto para el enlace *UpLink* como para *Downlink* provenientes de nodos finales a servidores de red e instrucciones desde servidores de red hacia los nodos finales. Los reenviadores de paquetes (*packet forwarders*) se pueden clasificar en monocales y

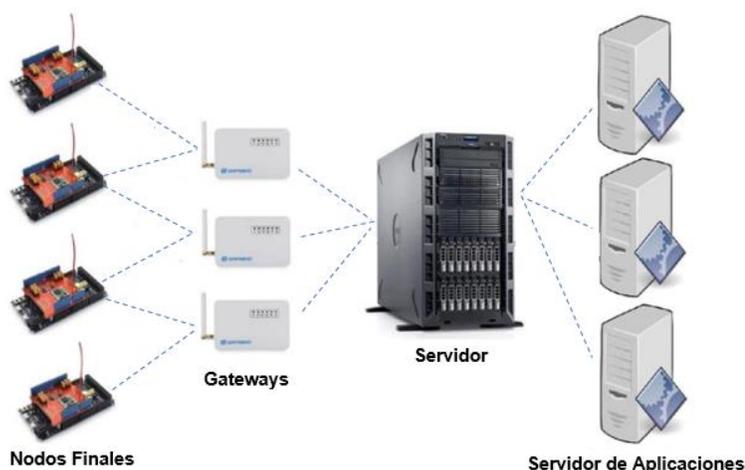
multicanales, presentan compatibilidad con LoRaWAN únicamente los reenviadores multicanal. Generalmente, un reenviador de paquetes realiza las siguientes funciones:

- Realiza el reenvío de paquetes LoRa recibidos por el Gateway o módulo LoRa hacia el servidor de red por medios de protocolos IP/UDP.
- Emite los paquetes LoRa que son transmitidos por el servidor de red.

El servidor de la red se encarga de enviar y recibir los mensajes LoRaWAN desde y hacia diferentes dispositivos, así como también con los servidores de aplicaciones. La carga útil en los mensajes LoRaWAN es enviado hacia el servidor de aplicaciones (Seneviratne, 2019).

### Figura 18

*Elementos de la red LoRaWAN*



*Nota.* La figura muestra el esquema básico del proceso de transmisión entre los componentes principales de la red LoRaWAN. Tomado de *Elements of LoRaWAN* (p.23) por P, Seneviratne, 2019, APress.

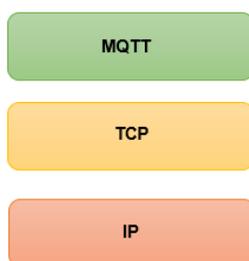
## **MQTT**

Para realizar el proceso de comunicación e intercambio de información entre dispositivos de una red se puede utilizar HTTP y construir el modelo *publish/subscribe*. No obstante, existe un estándar diseñado a ser mucho más ligero que HTTP, que funciona de manera eficiente en redes inalámbricas y con requerimientos de conectividad intermitente. La opción más apropiada para escenarios en los cuales se requiere intercambiar datos entre dispositivos casi en tiempo real por medio de Internet y consumir el menor ancho de banda posible es MQTT (Hillar, 2017).

MQTT constituye un estándar abierto de mensajería desarrollado por OASIS, orientado a IoT y una recomendación de la ISO (ISO/IEC 20922). Diseñado para ser un transporte de mensajes de publicación/suscripción, de carácter ligero e ideal para la conexión de dispositivos remotos y un óptimo consumo de ancho de banda. MQTT es ampliamente utilizado en gran cantidad de sectores de producción como: automotor, telecomunicaciones, sistemas de monitorización petroleros y gas, aplicaciones de chat, servicios de notificación, etc. La Figura 19, muestra el protocolo MQTT sobre la pila TCP/IP (Hillar, 2017).

### **Figura 19**

*Protocolo MQTT*



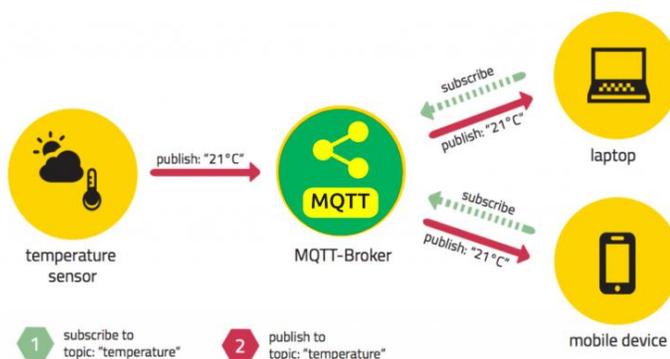
*Nota.* La figura muestra la ejecución de MQTT sobre TCP/IP. Tomado de *MQTT protocol on top of the TCP/IP stack* (p.28) por G, Hillar, 2017, Packt.

MQTT está basado en el patrón *Publish-Subscribe* que separa a los clientes los mismos que envían mensajes sobre temas específicos llamados tópicos (*topics*) hacia otro cliente o más clientes que reciben la información siempre y cuando estén suscritos. El cliente no sabe de la inexistencia de los otros clientes que conforman la red, publican mensajes de un tipo en específico y únicamente los clientes que están interesados en definidos tipos de mensajes reciben los mensajes que son publicados (Hillar, 2017).

*Publish-Subscribe* requiere del uso de un *bróker*, también denominado como servidor al cual requieren realizar una conexión (suscripción) todos los dispositivos (clientes). *Publisher*, es un cliente que envía mensajes bajos ciertos tópicos al bróker, el cual se encarga del filtrado de los mensajes que ingresan y de la distribución a los dispositivos (suscriptores) que se suscriben al tópico. La Figura 20, presenta un diagrama de 2 suscriptores conectados al *bróker* y un *publisher* (Hillar, 2017).

**Figura 20**

*Diagrama Publish-Subscribe*



*Nota.* Tomado de *Pub/sub approach*, por Manzoni. P (2019, p.7)

## Tópicos

Un tópico hace referencia a una cadena de caracteres que el *bróker* usa para realizar el proceso de filtrado de los mensajes transmitidos por cada cliente conectado.

Los tópicos de MQTT se encuentran estructurados con una jerarquía similar a la de carpetas y archivos en los sistemas operativos, mediante el uso de la barra diagonal (/) se delimita y facilita el entendimiento de la información que se transmite, de esta manera es posible crear estructuras con una nomenclatura fácil de utilizar y de carácter autodescriptivo (Manzoni, 2019). Los nombres de los tópicos deben utilizar cadenas de tipo UTF-8, estar constituido de por lo menos un carácter para ser válido y distinguir entre mayúsculas o minúsculas, como se presenta en la Figura 21.

### Figura 21

*Nomenclatura de los Tópicos*

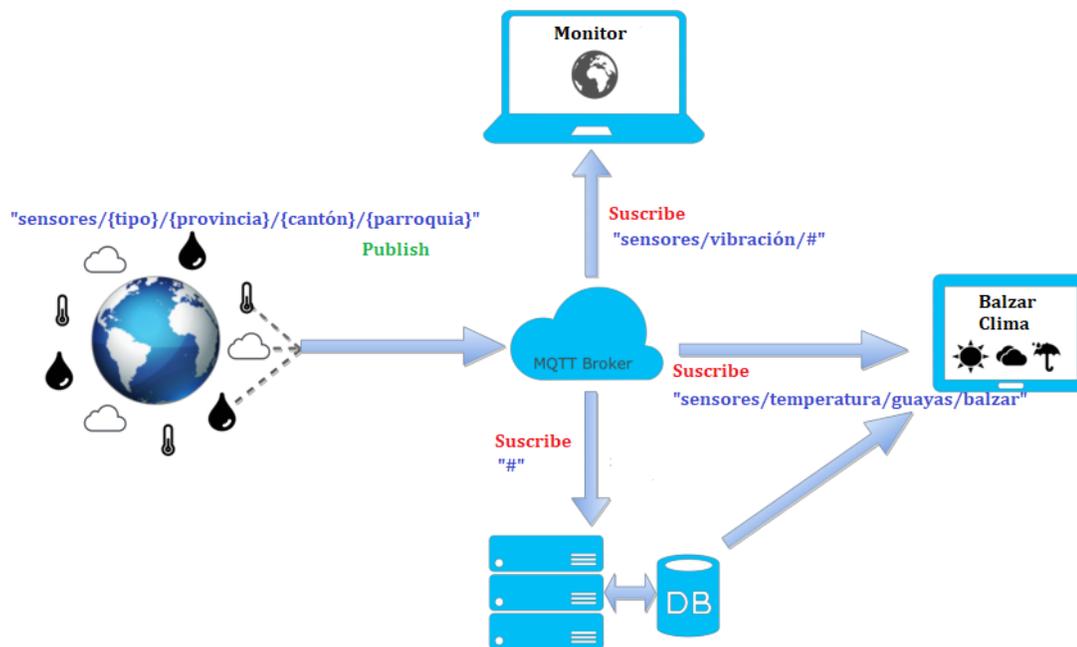


*Nota.* Tomado de *Naming structures*, por Manzoni. P (2019, p.7)

Las subscripciones a los tópicos pueden realizarse por medio de comodines, estos posibilitan a los nodos suscribirse a grupos de tópicos y aumentar la flexibilidad en la estructura de mensajería por toda la red. '+', constituye un comodín de tipo mono nivel que sustituye a un nivel del tópico, mientras que con el comodín '#', se suscribe a todos los niveles del tópico. En la Figura 22, se realiza la representación del proceso de comunicación y transmisión de información bajo el sistema *Publish-Subscribe*.

**Figura 22**

*Esquema Publish, Subscribe y Bróker*



*Nota.* La figura muestra el esquema básico del proceso de transmisión bajo el sistema Publish-Subscribe. Para recibir los datos del bróker MQTT de manera directa en los navegadores web, es necesario el uso de WebSockets. TCP y WebSockets operan sobre la capa Transport Layer Security (TLS) y su predecesor, Secure Sockets Layer (SSL). Tomado de Intro to MQTT, por Manzoni. P (2019, p.8)

Basado en el esquema de la Figura 22, al suscribirse al tópic "sensores/#" se recibe la información de todos los tipos de sensores instalados en las diversas provincias, cantones y parroquias, mientras que al usar el comodín '+' y suscribirse al tópic *sensores+/Guayas* se recibe la información de todos los tipos de sensores que se hallen instalados únicamente en la provincia del Guayas.

### **Calidad de Servicio**

La información transmitida se publica con un determinado nivel de calidad de servicio (QoS, del inglés *Quality of Service*), que define los requisitos en la entrega de los datos.

- QoS igual a 0, basa su funcionamiento en TCP en el cual se entregan los mensajes únicamente a los dispositivos (clientes) que se hallen suscritos.
- QoS igual a 1, permite asegurar que los mensajes lleguen a los suscriptores, pero no garantiza que no sean duplicados.
- QoS igual a 2, asegura y verifica que los mensajes transmitidos lleguen una sola vez a los suscriptores.

### **Topologías de Red**

Una topología de red se define como la disposición lógica o física empleada para el proceso de comunicación e intercambio de información entre los distintos nodos, detalla la manera en que los diferentes nodos finales se encuentran conectados entre sí, en la cual elijan la ruta más adecuada para la transmisión (Ortiz, 2020).

En el diseño de redes inalámbricas LPWAN, la topología es un factor indispensable, debido a que está directamente relacionado con parámetros como el retardo, potencia, velocidad de transmisión y redundancia. Las topologías de red más utilizadas son la estrella y malla, aunque existen una gran diversidad de topologías de red que van a ser descritas a continuación, cada una con distintas ventajas y desventajas.

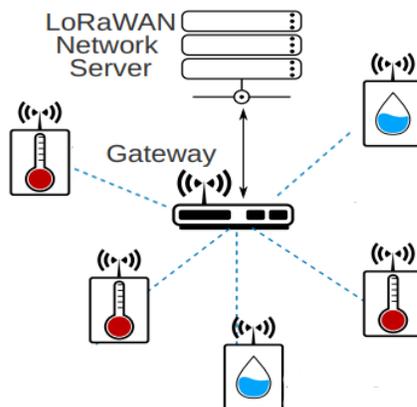
#### ***Topología de red estrella***

Una red con topología de estrella está compuesta de un nodo *Gateway* al cual se conectan todos los nodos finales y que se encarga de transmitir la información al

servidor central. Para comunicarse entre sí, los nodos requieren del *Gateway*, el mismo que se encarga de retransmitir los mensajes recibidos de los nodos finales al servidor de la red, en el cual se llevan a cabo tareas de comprobación de redundancia, errores y seguridad. La característica de salto único (*single-hop*) de las redes en estrella hacen que sean rápidas y confiables. La ventaja de esta topología es la fácil identificación de nodos defectuosos para aislarlos instantáneamente, pero si el *Gateway* falla, se produce la inaccesibilidad de todos los nodos. La tecnología LPWAN LoRa admite la topología en estrella, como se presenta la Figura 21.

### Figura 23

*Red con topología estrella*



*Nota.* La imagen muestra la topología estrella en una red LoRaWAN, la cual está formada por un *Gateway* al que transmiten información los nodos finales. Tomado de LoRaWAN network, por Rodríguez. J & Kleinschmidt. J (2020, p.15).

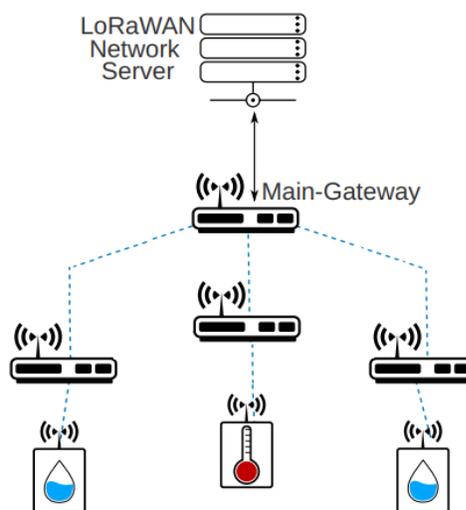
### **Topología de red en árbol**

La topología está basada en jerarquía de nodos y en la combinación de las topologías en estrella y bus. El nodo central, consiste en el nivel jerárquico más alto de la red en árbol al cual están conectados varios nodos finales por medio de conexiones punto a punto. Los nodos inferiores transmiten la información hacia el *Gateway* central

por medio del nodo principal de cada enlace. Este tipo de topología es propensa a padecer fallas en la red cuando el nodo central sufre algún daño (Peter & Perttunen, 2014).

### Figura 24

*Red con topología en árbol*



*Nota.* La imagen ilustra la topología en árbol en una red LoRaWAN compuesta de los *Gateways* y nodos finales encargados de la recolección y transmisión de datos. Tomado de LoRaWAN network, por Rodríguez. J & Kleinschmidt. J (2020, p.15).

### **Topología de red malla**

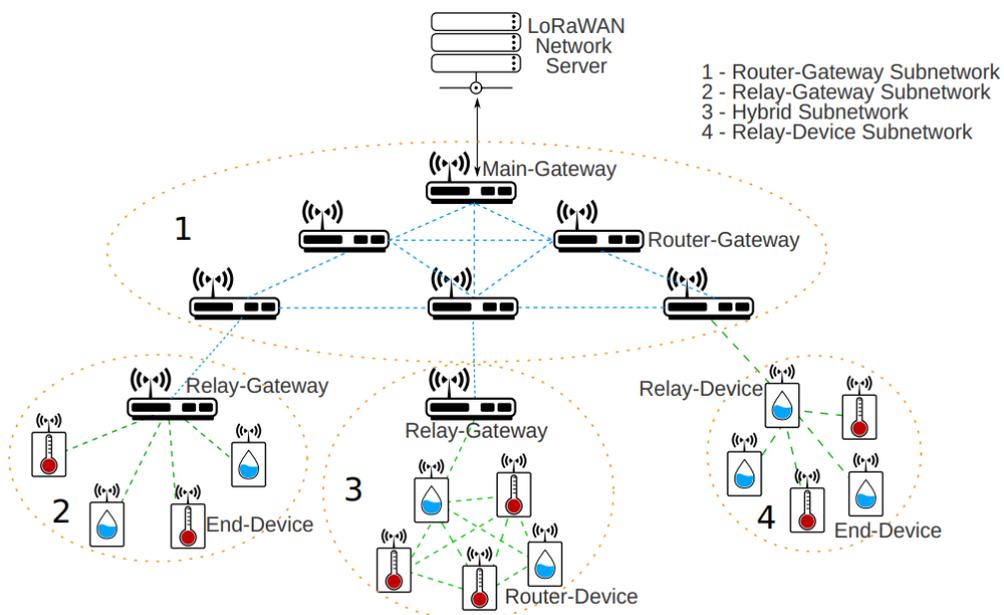
La topología tipo estrella puede modificarse a varios tipos de redes con la inserción de un dispositivo intermedio. En la Figura 25, se hace una representación de las diversas configuraciones de redes con topologías tipo malla. La subred 1 representa el proceso de comunicación entre *Gateways* mediante el uso de protocolos, el *Gateway* principal se encuentra conectado al servidor de la red y los demás son *Gateways* de enrutamiento. En la subred 2, se utiliza un *Gateway* de retransmisión para realizar la conexión a los dispositivos y nodos finales. La subred 3 es una red híbrida pues

presenta una solución con dispositivos de enrutamiento tipo malla y un *Gateway* de retransmisión. Tanto en la subred 2 y 3, existe un *Gateway* que se encarga de realizar la conexión de los dispositivos y nodos remotos con el *Gateway* del nivel de red (subred 1). Finalmente, en la subred 4 el dispositivo final se encarga de retransmitir la información y al mismo tiempo del proceso de recolección de datos (Rodríguez & Kleinschmidt, 2020).

Las ventajas que ofrecen las redes con topología de malla son la existencia de múltiples rutas de acceso, transmisiones simultáneas *up/downstream* y escalabilidad de la red, no obstante, las desventajas de estas redes son el incremento de los costos de implementación, aumento de la complejidad por la existencia de nodos redundantes, la latencia ocasionada por comunicaciones multisalto (*multihop*) y alto consumo energético (Chaudhari & Zennaro, LPWAN Technologies for IoT and M2M Applications, 2020).

**Figura 25**

*Red con topología de malla*



*Nota.* La imagen ilustra la topología de malla en una red LoRaWAN y los dispositivos que conforman las subredes. Tomado de LoRaWAN mesh network, por Rodríguez. J & Kleinschmidt. J (2020, p.13).

### ***Topología Repetidor***

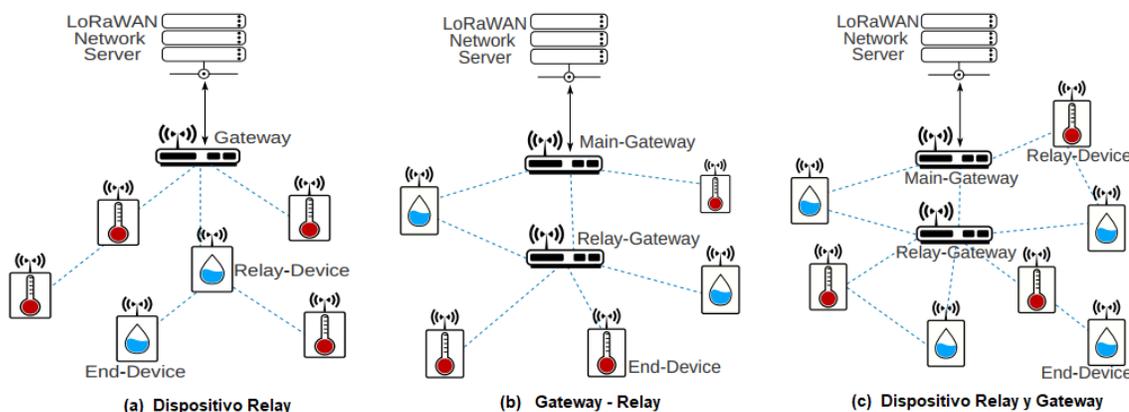
Los dispositivos repetidores son utilizados con la finalidad de proporcionar o mejorar la conexión de nodos remotos. La Figura 26(a) representa el uso de dispositivos que funcionan como repetidor y como un nodo final. Esta topología es adecuada para soluciones como edificios inteligentes o aplicaciones industriales donde exista una fuente de alimentación. La Figura 26(b) expone una topología con *Gateway* repetidor que emplean un radio de tipo multicanal para escuchar un mínimo de ocho frecuencias, esto permite al *Gateway* repetidor focalizar el tráfico de los nodos finales remotos sin la necesidad de mecanismos de selección de la frecuencia empleada para la transmisión. Este tipo de topología es útil en escenarios rurales, para conectar los nodos remotos al *Gateway* principal mediante un *Gateway* repetidor. La diferencial principal entre la topología con dispositivos repetidores y la de *Gateway* repetidores radica en la fuente de alimentación. Dispositivos repetidores deben estar alimentados por baterías mientras que *Gateway* repetidor no tiene limitaciones energéticas (Rodríguez & Kleinschmidt, 2020).

La Figura 26(c) muestra la topología compuesta por Dispositivos repetidores y *Gateways* repetidores, un ejemplo del escenario de aplicación de esta arquitectura son las ciudades inteligentes, donde se requiere una red de retransmisión mixta es decir conectar nodos finales remotos y focalizar el tráfico en distintos puntos de la red. El Dispositivo repetidor permite realizar conexiones con nodos finales remotos y el *Gateway* repetidor permite concentrar los datos provenientes de un conjunto de nodos

finales y también de Dispositivos repetidores para transmisión de información hacia el *Gateway* principal (Rodríguez & Kleinschmidt, 2020).

**Figura 26**

*Topologías con nodos repetidores*



*Nota.* La imagen ilustra diferentes arquitecturas de topología de red con nodos relay.

Tomado de Architecture with relay nodes, por Rodríguez. J & Kleinschmidt. J (2020, p.14).

### **Topología de Enrutamiento**

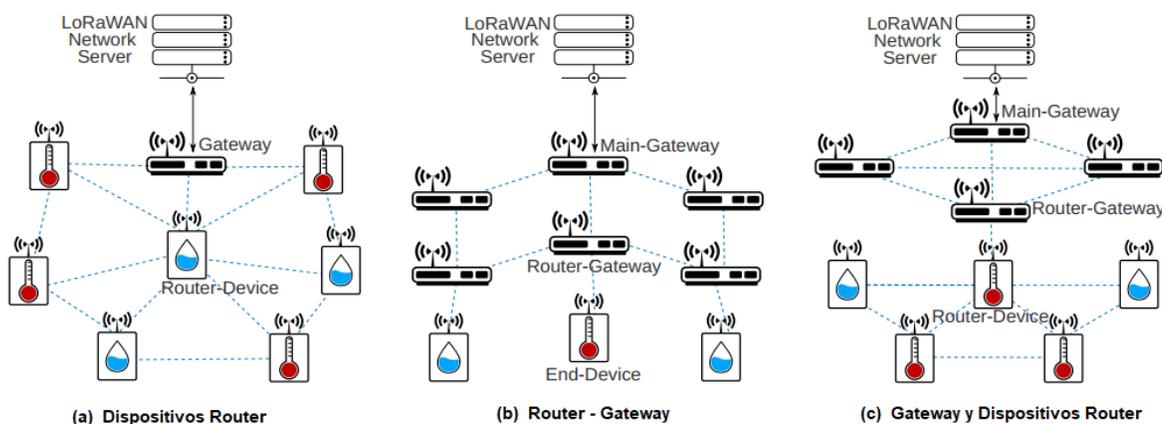
Esta topología detalla las múltiples configuraciones de red fundamentadas en dispositivos con capacidad de enrutamiento. Los protocolos de configuración pueden operar en los nodos finales o en el *Gateway*. La Figura 27, plantea 3 tipos de configuraciones para situaciones distintas. Para escenarios orientados a edificios inteligentes en los que existen gran cantidad de nodos en una red compleja con acceso a fuentes de alimentación y presencia de interferencias, la topología presentada en la Figura 27(a) es la más apropiada, debido al menor coste de los dispositivos router en comparación con un *Gateway* (Rodríguez & Kleinschmidt, 2020).

La Figura 27(b) presenta una topología compuesta por un sólo *Router-Gateway* al cuál están conectado los nodos finales. No obstante, es probable que el nodo final tenga conexiones con más de un *Router-Gateway* simultáneamente. Los *Router-Gateway* son los encargados de gestionar los paquetes repetidos en la red con el propósito de evitar la congestión de tráfico de datos innecesaria, trabajar como un concentrador de paquetes de datos provenientes de distintos dispositivos finales y establecer la mejor ruta para transmitir la información al servidor por medio del *Gateway* principal. Un ejemplo de aplicación de esta topología es la agricultura inteligente, para monitorizar variables físicas o controlar el proceso de riego (Rodriguez & Kleinschmidt, 2020).

La Figura 27(c) presenta la interconexión existente entre diferentes niveles de enrutamiento, un nivel para los dispositivos *Router* y otro para los *Router-Gateway*. La conexión de uno o varios dispositivos *Router* con uno o varios *Router-Gateways* viene definido por las limitaciones físicas, el gestor de red y protocolos de enrutamiento.

**Figura 27**

*Topologías con nodos router*



*Nota.* La imagen ilustra diferentes arquitecturas de topología de red con nodos router y la configuración mixta. Tomado de *Architecture with router nodes*, por Rodríguez. J & Kleinschmidt. J (2020, p.14).

### ***Elección de la topología de red***

**Tabla 10**

*Comparación de Topologías de red*

<b>Parámetro</b>	<b>Estrella</b>	<b>Malla</b>	<b>Árbol</b>
<b>Instalación</b>	Fácil	Difícil	Difícil
<b>Costo</b>	Moderado	Elevado	Elevado
<b>Flexibilidad</b>	Si	No	No
<b>Fiabilidad</b>	Alta	Alta	Alta
<b>Robustez</b>	Si	Si	No
<b>Consumo energético</b>	Moderado	Alto	Alto

*Nota.* Análisis comparativo entre las principales topologías de red para el despliegue de redes LPWAN LoRa. Tomado de *Analysis of Different Topologies*, por Lim. F (2016, p.180).

Existen muchos parámetros a considerar cuando se elige la topología adecuada para desarrollar una red de sensores inalámbricos, como se presenta en la Tabla 10. Por lo general para redes LPWAN se opta por la topología en estrella, puesto que permite la escalabilidad de la red con la conexión de una gran cantidad de nodos, reducción del consumo de energía, alta fiabilidad, costo moderado e incremento del alcance de transmisión. Por consiguiente, para el presente proyecto se implementa esta topología, la cual está constituida de un nodo final que será el encargado del proceso de monitorización, recopilación y transmisión directa de los datos al *Gateway*,

que reenvía los mensajes recibidos al servidor de AWS. En la Tabla 11 se describe las características del enlace UpLink y Downlink de LoRa:

**Tabla 11**

*Características del enlace ascendente y descendente*

<b>Característica</b>	<b>UpLink</b>	<b>Downlink</b>
<b>Modulación</b>	CSS	CSS
<b>Presupuesto del enlace</b>	156 dB	164 dB
<b>Tasa de bits (adaptativa)</b>	0.3 a 5 kbps	0.3 a 5 kbps
<b>Tamaño del mensaje por payload</b>	0 a 250 bytes	0 a 250 bytes
<b>Duración del mensaje</b>	40ms a 1.2s	20 a 160 ms

*Nota.* La tabla muestra características presentes en el enlace ascendente y descendente de LoRa. Tomado de *LPWAN Technologies for IoT and M2M Applications*. (p.6) por Bharat .C & Zennaro .M, 2020, Academic Press

### **Análisis de prestaciones técnicas para aplicaciones con LoRa**

Existen diferentes requisitos técnicos para las múltiples aplicaciones *LPWAN* establecidas en la Tabla 3, entre las principales están: cobertura, capacidad, costo y funcionamiento con bajo consumo. Cualquier solución *LPWAN* realiza diferentes compensaciones entre requisitos como, por ejemplo, cobertura con respecto al costo. De igual manera, algunas aplicaciones necesitan de otras capacidades como poder realizar interfuncionamiento con otras tecnologías de comunicación o soporte de transmisión de voz. En consecuencia, una tecnología *LPWAN* específica puede adaptarse a un número reducido de aplicaciones, de manera que otra puede diseñarse para cubrir las necesidades de un conjunto de aplicaciones y necesidades (Chaudhari & Zennaro, *LPWAN Technologies for IoT and M2M Applications*, 2020).

En la Tabla 12, se establece el énfasis a los requisitos técnicos requeridos en cada solución LPWAN con LoRa según la aplicación correspondiente. Aparte de los requisitos primarios como cobertura, capacidad, costo y funcionamiento con bajo consumo energético, se añade una categoría más denominada "especificación adicional", que cubre características complementarias (por ejemplo, transmisión de voz o interoperación con otra tecnología LPWAN) requeridas para una aplicación en particular. Las escalas relativas en las que se define los requerimientos de aplicabilidad son Alta (H), Media (M), Baja (L) y proporciona un contexto de solución que facilite la elección arquitectónica y de diseño más adecuada en base a la aplicación o conjunto de aplicaciones a los que se direcciona.

**Tabla 12**

*Métricas de requisitos de aplicabilidad para aplicaciones IoT con LoRa*

<b>Aplicación</b>	<b>Cobertura</b>	<b>Capacidad</b>	<b>Costo</b>	<b>Bajo Consumo</b>	<b>Especificación adicional</b>
<i>Smart cities</i>	H	H	M	M	H
<i>Smart environment</i>	M	H	M	H	H
<i>Smart water</i>	H	M	M	M	L
<i>Smart metering</i>	H	H	H	M	L
Red y energía inteligente	H	H	M	M	M
Seguridad y emergencias	H	L	M	H	H
Automóviles y logística	H	H	H	L	H
Agricultura inteligente	H	H	H	H	L
Hogares inteligentes	H	M	H	L	L
Salud, ciencias de la vida	H	H	M	H	H

*Nota.* La tabla muestra la cuantificación de los principales requerimientos para poner en funcionamiento aplicaciones del mundo real con tecnología LoRa. Tomado de *LPWAN Technologies for IoT and M2M Applications*. (p.6) por Bharat .C & Zennaro .M, 2020, Academic Press

### **Módulos de LoRa**

Para optimizar el manejo e implementación para soluciones de tipo IoT, los transceptores de LoRa desarrollados por Semtech son integrados en placas. En el mercado de telecomunicaciones existen varios módulos de LoRa, entre los más populares se encuentran: Reyax RYLR869, Dragino LoRa, MKR WAN 1300, TTGO LoRA 32. Para el desarrollo del proyecto se utiliza el módulo LoRa Shield y el *Gateway* de Dragino LG01N.

### ***LG01N LoRa Gateway***

LG01N es un *Gateway* de código abierto y de un sólo canal, mediante WiFi, Ethernet o una red celular 3G/4G permite la conexión de la red inalámbrica LoRa a Internet. Proporciona comunicación con espectro ensanchado de largo alcance y resistencia alta frente a interferencias. Puede ser utilizado para otorgar soluciones inalámbricas para aplicaciones IoT con bajo costo y soportar hasta 100 nodos sensores. El *Gateway* también soporta múltiples protocolos de comunicación como: MQTT, cliente TCP/IP y LoRaWAN para ajustarse a los requisitos de conexión y transmisión de datos (Dragino, 2020), presenta las siguientes características:

Hardware:

- Procesador ar9331 400MHz
- Memoria RAM de 64MB
- Memoria Flash de 16MB

**Interface:**

- Entrada de alimentación: 12V DC
- 2 puertos RJ45 con 10M/100m
- Estándar WiFi:802.11b/g/n

**Especificaciones LoRa:**

- Rango de Frecuencia: 862 a 1020 MHz
- Presupuesto máximo de enlace de 68 dB.
- Salida de RF constante de +20 dBm con 100 mW.
- PA de alta eficiencia de +14 dBm.
- Tasa de bits programable hasta 300 kbps.
- Alta sensibilidad: hasta -148 dBm.
- Baja corriente de RX de 10,3 mA, retención de registro de 200 nA.
- Modulación FSK, GFSK, MSK, GMSK, LoRa TM y OOK.

**Figura 28***Dragino LG01 Gateway*

*Nota.* La figura muestra el *Gateway* utilizado para el desarrollo de la red de sensores.

Tomado de LG01N/OLG01N LoRa *Gateway* (p.1) por Dragino, 2020.

### ***LoRa Shield***

Para el proyecto se emplea a Dragino LoRa Shield como transceptor para el envío de datos a larga distancia, presenta compatibilidad con Arduino Leonardo, Uno, Mega, DUE y con grandes prestaciones para proyectos de sensores inalámbricos con transmisiones en tiempo real como ciudades inteligentes, sistema de riego, monitorización, etc. Dispone de alta sensibilidad, un amplificador de potencia de +20 dBm, alta inmunidad a la interferencia, bajo consumo de energía (Tabla 13), frecuencia de operación de 915MHz y antena externa conectada por medio de un conector I-Pex, el transmisor presenta las siguientes características: (Dragino, 2020)

- Presupuesto de enlace de 168dB
- Nivel alto de sensibilidad de hasta -148 dBm
- Alta inmunidad a la interferencia
- Rango dinámico RSSI de 127 dB
- Paquetes de datos de hasta 256 bytes con código de redundancia cíclica
- Incorporación de sensor de temperatura e indicador de nivel de batería baja.
- Salida de Radiofrecuencia constante de +20 dBm a 100 mW

### **Figura 29**

*Dragino LoRa Shield*



*Nota.* La figura muestra el transmisor utilizado para el desarrollo de la red de sensores, acoplado a la placa Arduino UNO. Tomado de LG01N/OLG01N LoRa Shield (p.1) por Dragino, 2020.

En la Tabla 13 se expone el consumo de energía en los principales modos de operación:

**Tabla 13**

*Especificaciones de consumo de energía*

<b>Modo</b>	<b>Símbolo</b>	<b>Corriente de alimentación</b>	<b>Unidad</b>
<b>Reposo "Sleep"</b>	IDSSL	0.2	uA
<b>Idle</b>	IDDIDLE	1.5	uA
<b>Recepción</b>	IDDR	12	mA
<b>Transmisión</b>	IDDT	120 con +20dBm	mA
		87 con +17dBm	
		29 con +13dBm	
		20 con +7dBm	

*Nota.* La tabla realiza una descripción del consumo energético en los diferentes modos de operación del equipo. Tomado de LG01N/OLG01N LoRa Shield (p.1) por Dragino, 2020.

### **Sensores para monitorización remota**

La monitorización remota combinada con el uso de sensores que transmiten información en tiempo real e instalados de manera adecuada en el contenedor de la solución nutritiva y en los semilleros del sistema hidropónico permiten realizar la medición precisa de variables como el pH, la humedad, temperatura y el nivel de agua. A continuación, se exponen los sensores seleccionados para el desarrollo de la red.

### **Sensor de Temperatura**

Un sensor de temperatura constituye un dispositivo electrónico que provee la medición de valores de energía térmica de un objeto o sistema por medio de una señal de tipo eléctrica y que genera una salida de tipo analógica o digital (Corona, Abarca, & Mares, 2014). Antes de definir el tipo de sensor de temperatura más adecuado para el proyecto, en la Tabla 14 se realiza una comparativa de las prestaciones entre los principales sensores utilizados en proyectos electrónicos.

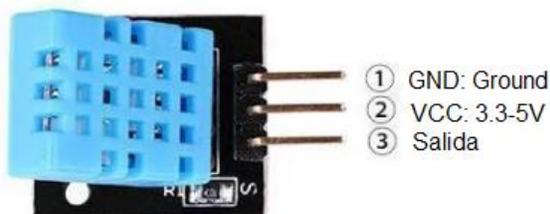
**Tabla 14**

#### *Sensores de Temperatura*

<b>Sensor</b>	<b>Nivel de Exactitud</b>	<b>Nivel de Resolución</b>	<b>Rango de Temperatura</b>	<b>Costo</b>
<b>LM35</b>	$\pm 2 \text{ }^{\circ}\text{C}$	10mV/ $^{\circ}\text{C}$	-55 a 50 $^{\circ}\text{C}$	Bajo
<b>DHT11</b>	$\pm 2 \text{ }^{\circ}\text{C}$	8 bits	0 a 60 $^{\circ}\text{C}$	Bajo
<b>DHT22</b>	$\pm 0.5 \text{ }^{\circ}\text{C}$	12 bits	-40 a 80 $^{\circ}\text{C}$	Medio

*Nota.* La tabla realiza una comparación entre los sensores de temperatura más populares en el mercado de electrónica. Tomado de Ventajas y desventajas de sensores de temperatura. (p.175) por Corona. L, Abarca. G & Mares. J, Editorial Patria.

Luego de realizar el análisis de alternativas, se escoge al sensor DHT11 por su nivel de exactitud, costo, flexibilidad de uso, calibración estricta de la señal de salida por los fabricantes, alta fiabilidad, óptima estabilidad, bajo consumo energético y al disponer de un rango de medición de 0 a 60  $^{\circ}\text{C}$  que se ajusta a los requisitos del proyecto. El sensor DHT11 es empleado para medir la temperatura en los semilleros de germinación, dispone de 3 pines: Conexión de VCC, GND y la salida de datos, como se expone en la Figura 30.

**Figura 30.***Sensor DHT11*

*Nota.* Disposición de conexión de los pines del sensor DHT11. Tomado de Arquitectura del sensor de temperatura DHT11. (p.175) por Corona. L, Abarca. G & Mares. J, Editorial Patria.

Cerámicas semiconductoras o polímeros conforman al sensor DHT11 el cual está integrado de un sensor de humedad capacitivo y de un termistor con coeficiente de temperatura negativo que al producirse un aumento de temperatura disminuye su valor. El sensor presenta una frecuencia de muestreo de 1Hz equivalente a 1 muestra por segundo, al cual se debe añadir una resistencia *pull-up* con valores de 5k a 10k ohmios para realizar la correcta comunicación entre el sensor y el microcontrolador. La Tabla 15 Especificaciones Técnicas DHT11, muestra las especificaciones definidas por el fabricante (Corona, Abarca, & Mares, 2014).

**Tabla 15***Especificaciones Técnicas DHT11*

Parámetro	Especificación
<b>Alimentación</b>	3.5 a 5 V
<b>Sensibilidad</b>	Humedad: 1% Temperatura: 1°C
<b>Exactitud</b>	Humedad: ±5% Temperatura: ±2°C

Parámetro	Especificación
Corriente	0,3mA (midiendo) 60uA (en espera)
Rango	Temperatura: 0 a 60 °C, Humedad: 20% a 90%

*Nota.* Tomado de DHT11 sensor specifications datasheet. Mouser Electronics (2021).

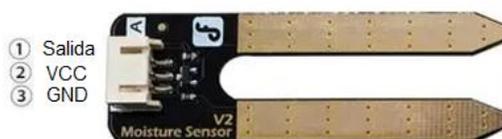
### **Sensor de Humedad**

Para la germinación adecuada de las plantas es indispensable realizar la monitorización y recolección de parámetros de humedad de una manera precisa, con el objetivo de tomar una decisión oportuna y controlar este parámetro en base a los resultados de las mediciones obtenidas En (Corona, Abarca, & Mares, 2014), manifiesta que hay muchas formas de representar la humedad, en general esta es representada en humedad relativa, específica y absoluta. La humedad absoluta expresa la relación existente entre el número de moléculas de agua existentes en una sustancia y su volumen. La humedad relativa relaciona la cantidad de agua contenida en un gas y en estado de saturación la cantidad de gas sin que exista condensación. La humedad específica denominada también como contenido de humedad, relaciona la masa de agua y masa de sustancia seca (Corona, Abarca, & Mares, 2014).

Para el proyecto, por ser la opción más popular en cuanto a costos, disponibilidad en el mercado electrónico y un elevado nivel de fidelidad se hace uso del sensor de estado sólido SEN0114.

### **Figura 31**

*Sensor de Humedad SKU: SEN0114*



*Nota.* Disposición de conexión de los pines del sensor SEN0114. Tomado de Arquitectura del sensor de humedad. (p.211) por Corona. L, Abarca. G & Mares. J, Editorial Patria.

SEN0114 es un sensor de humedad analógico que hace uso de dos sondas para hacer circular corriente eléctrica a través del suelo, luego interpreta la resistencia y adquiere el grado de humedad relativo. Un suelo húmedo conduce con mayor facilidad la electricidad debido a que existe menos resistencia, en comparación a suelos secos donde el nivel de resistencia es mayor por la poca cantidad de agua. La Tabla 16, muestra las especificaciones definidas por el fabricante (Corona, Abarca, & Mares, 2014).

**Tabla 16**

*Especificaciones Técnicas SKU: SEN0114*

<b>Parámetro</b>	<b>Especificación</b>
<b>Alimentación</b>	3.3 a 5 V
<b>Salida</b>	0 a 4.2 V
<b>Corriente</b>	35 mA
<b>Rango en suelo seco</b>	0 a 300
<b>Rango en suelo húmedo</b>	300 a 700
<b>Rango en agua</b>	700 a 950

*Nota.* Tomado de SEN0114 sensor *specifications datasheet*. Sigma Electronics (2021).

### **Sensor Ultrasónico**

Un sensor ultrasónico pertenece a la categoría de sensores de nivel y proximidad, orientados a sistemas de medición de tipo no invasivo para determinar la proximidad de cualquier objeto en relación al sensor o establecer el porcentaje de nivel

de un contenedor. Al tratarse de mediciones no invasivas, no es necesario el contacto para efectuar la medida y debido a las características de una señal ultrasónica es posible ejecutar mediciones en diversas superficies, líquidos y ambientes con condiciones hostiles (Corona, Abarca, & Mares, 2014). En el proyecto se emplea al sensor ultrasónico HC-SR04 presentado en la Figura 32, para medir el nivel de solución nutritiva en el contenedor.

### Figura 32

*Sensor Ultrasónico HC-SR04*



*Nota.* Disposición de conexión de los pines del sensor HC-SR04. Tomado de Descripción del sensor HC-SR04. (p.158) por Corona. L, Abarca. G & Mares. J, Editorial Patria.

HC-SR04 es un sensor ultrasónico constituido de dos transductores, el primero opera como un transmisor que se encarga de convertir la señal eléctrica en pulsos de ultrasonido a una frecuencia de 40 kHz. El otro transductor se encarga de escuchar los pulsos transmitidos, cuando los recibe genera un impulso de salida cuyo ancho es utilizado para establecer la distancia recorrida por el pulso ultrasónico. La Tabla 17, muestra las especificaciones técnicas de operación del sensor entregadas por el fabricante (Corona, Abarca, & Mares, 2014).

**Tabla 17***Especificaciones Técnicas HC-SR04*

<b>Parámetro</b>	<b>Especificación</b>
<b>Alimentación</b>	5 V
<b>Corriente de operación</b>	15 mA
<b>Frecuencia de operación</b>	40 KHz
<b>Alcance máximo</b>	4m
<b>Alcance mínimo</b>	2cm
<b>Precisión</b>	3mm

*Nota.* Tomado de HC-SR04 Ultrasonic Sensor Specification Sheet. elecbreaks (2018).

### **Sensor de pH**

Un sensor de pH constituye un transductor entre una señal de tipo eléctrica y los niveles de acidez de una solución. Estos sensores son de gran utilidad en procesos industriales, al ser indispensable monitorear y controlar el rango de pH en soluciones usadas en procedimientos químicos, tales como, elaboración de bebidas, alimentos, productos derivados del petróleo y procesos de tratamiento de residuos de sustancias corrosivas peligrosas (Corona, Abarca, & Mares, 2014).

La abreviatura pH hace referencia al nivel de potencial hidrógeno en una sustancia, al medir el pH se logra definir qué tan ácida o alcalina es una solución. El rango de variación de este parámetro tiene una escala con valores de 0 a 14, se considera que una sustancia es ácida cuando se tiene valores menores a 7, valores superiores a 7.1 determina que existe alcalinidad y una medida de 7 indica un valor neutro (Corona, Abarca, & Mares, 2014). El valor de pH se expresa de manera matemática por medio de la ecuación (8):

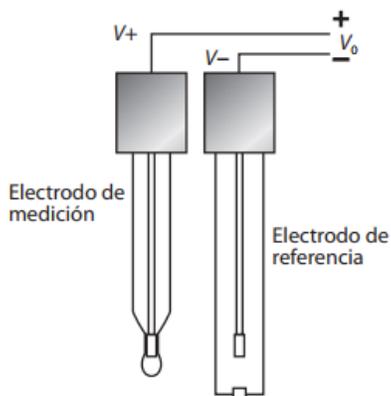
$$pH = \log_{10} a_{H^+} \quad (8)$$

Donde  $a_{H^+}$  expresa el grado de concentración molar de iones hidronio.

Para comprender el funcionamiento de un sensor de pH se toma en consideración lo siguiente: un pequeño potencial es generado en el proceso de contacto de un material metálico con una solución salina o ácida, lo mismo ocurre cuando dos sustancias líquidas entran en contacto, por lo que sólo se necesita utilizar una membrana para evitar que se mezclen por completo, como se observa en la Figura 33. El pequeño potencial recibe el nombre de potencial electroquímico y define la diferencia de iones a ambos lados de la membrana, que ocasiona diferencias tanto en la carga eléctrica como en la concentración química de las dos sustancias (Corona, Abarca, & Mares, 2014).

### Figura 33.

*Estructura para la medición de pH*



*Nota.* Configuración de electrodos para la medición de pH en sustancias líquidas.

Tomado de Arquitectura básica para la medición de pH. (p.274) por Corona. L, Abarca.

G & Mares. J, Editorial Patria.

En el mercado electrónico ecuatoriano no existe mucha diversidad en cuanto a sensores de pH, la opción más viable para el proyecto por su alta precisión y costo, es la sonda E201-BNC (Figura 34) integrada con la tarjeta de acondicionamiento PH-4502C (Figura 35), las características técnicas de estos dispositivos se muestran en la Tabla 18 y Tabla 19 respectivamente.

### Figura 34

*Sonda de pH E201-BNC*



*Nota.* Tomado de Liquid pH Sensor with BNC. Electrobes (2018).

### Tabla 18

*Especificaciones Técnicas Sonda E201-BNC*

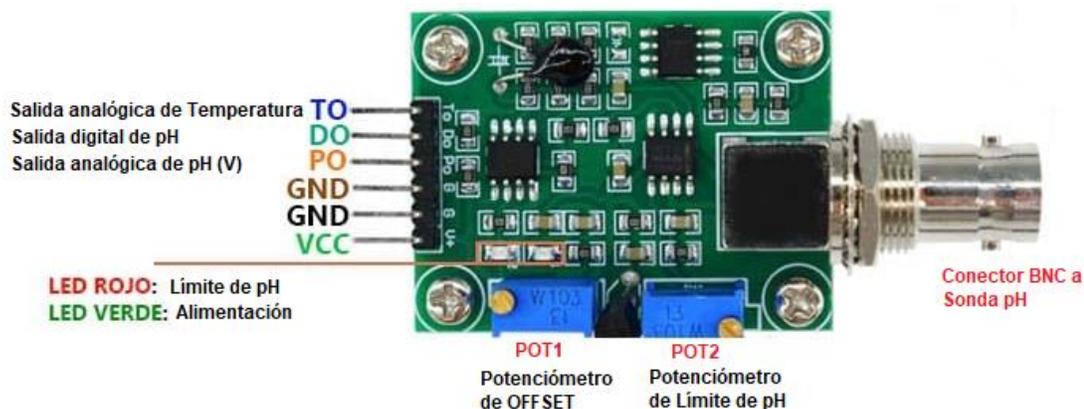
Parámetro	Especificación
Alimentación	5 V
Corriente de operación	5 a 10 mA
Temperatura de operación	10 a 50 °C
Tiempo de respuesta	≤ 5 s
Tiempo de establecimiento	≤ 60 s
Calibración	pH 7 a 25°C
Rango de medición pH	0 a 14 pH
Rango de temperatura	0 a 60 °C

*Nota.* Tomado de Liquid pH Sensor with BNC pH Electrode Specification Sheet. Electrobes (2018).

La sonda de pH está conformada de un medidor de temperatura y dos electrodos: sensitivo (un vidrio) y de referencia. El electrodo sensitivo sumergido en la solución, es el encargado de generar una tensión proporcional al valor de pH sentido, mientras que la función del electrodo de referencia es mantener la tensión fija independientemente de los cambios de temperatura. La diferencia entre la tensión del electrodo sensitivo y la del de referencia debe ser proporcional al pH de la solución. El fabricante generalmente calibra a las sondas de pH con un valor de 0 V equivalente a un nivel de pH de 7 a 25 °C. (Corona, Abarca, & Mares, 2014).

### Figura 35

*Tarjeta de acondicionamiento PH-4502C*



*Nota.* Disposición de conexión de los pines de la tarjeta de acondicionamiento de pH-4502C. Tomado de Conditioning card for pH probe. Electrobes (2017).

### Tabla 19

*Especificaciones Técnicas Tarjeta de Acondicionamiento 4502C*

<b>Parámetro</b>	<b>Especificación</b>
<b>Alimentación</b>	5 V
<b>Corriente de operación</b>	5 a 10 mA
<b>Temperatura de operación</b>	10 a 50 °C
<b>Tiempo de respuesta</b>	≤ 5 s
<b>Tiempo de establecimiento</b>	≤ 60 s
<b>Tipo de Salida</b>	Analógica
<b>Consumo de potencia</b>	≤ 5 W

*Nota.* Tomado de Liquid pH Sensor with BNC pH Electrode Specification Sheet. Electrobes (2018).

### **Servidor en la nube**

Para seleccionar la plataforma apropiada para el desarrollo de proyectos digitales se requiere conocer aspectos técnicos y costos para el despliegue óptimo de la red. La mayor parte de empresas y proyectos tienen una preferencia por *Data Centers*, estos centros físicos necesitan una gran inversión en temas de hardware, desarrollo de software y técnicos capacitados para realizar los procesos de instalación, configuración y mantenimiento. En base a lo expuesto, el costo de inversión en estos centros de datos son un impedimento en proyectos de poco presupuesto, por ello grandes compañías ofrecen servicios de tecnologías de virtualización para procesar y almacenar datos. El crecimiento de las tecnologías en la nube, hace posible el acceso a una gran cantidad de usuarios al poder del cómputo con infraestructuras de primer nivel y de bajo precio, desde grandes empresas hasta personas jóvenes que desde sus hogares diseñan aplicaciones de todo tipo (Noriega, 2021).

En el mercado existen tres grandes empresas que ofrecen servicios de Cloud Computing con diferentes funcionalidades, precios, productos y servicios. Estas empresas son: *Amazon Web Services (AWS)*, *Google Cloud Platform (GCP)* y *Microsoft Azure*, a continuación, se realiza una comparativa entre cada una de ellas.

### **Productos y Soluciones**

Una solución se define como el conjunto de productos que vienen preconfigurados y dirigidos a una necesidad definida, disponen de gran cantidad de documentación, ejemplos de uso y diversos testimonios que orientan a lo largo del proceso de la elección de la infraestructura en la nube. En la Tabla 20 se realiza una comparación entre las diversas soluciones ofrecidas por las empresas de Cloud Computing (Noriega, 2021).

**Tabla 20**

*Soluciones de Cloud Computing*

<b>Solución</b>	<b>AWS</b>	<b>GCP</b>	<b>Microsoft Azure</b>
<b>Amplio volumen de datos</b>	x	x	X
<b>Marketing digital</b>	x	x	X
<b>Comercio electrónico</b>	x	x	X
<b>Desarrollo de aplicaciones</b>	x	x	
<b>IoT</b>	x	x	X
<b>Nubes privadas</b>	x	x	X
<b>Hosting</b>	x	x	X

*Nota.* La tabla muestra una comparativa entre soluciones ofrecidas por las tres empresas más importantes de servicios en la nube, para sistemas del Internet de las

cosas Microsoft Azure realiza mantenimientos de carácter predictivo y control de manera remota. Tomado de productos y soluciones Por Noriega. A (2021)

### **Precios**

La estrategia de precios altamente competitivos y flexibles son las grandes ventajas que ofrecen los proveedores de servicios en la nube. En los inicios se realizaba un pago mensual según la solución empleada, no obstante, en la actualidad surge la tendencia del método de cobro por cada segundo de potencia en cómputo denominado como PAYG (del inglés, *Pay As You Go*) (Noriega, 2021). Existen tres aspectos primordiales de los servicios en la nube que establecen el precio de sus planes, como son:

- **Sistema Operativo:** El tipo de sistema operativo influye de manera directa en el costo del plan de servicio. Estos pueden ser gratuitos o de pago. Por ejemplo, el sistema operativo CentOS es gratuito en la mayoría de plataformas Cloud, a diferencia de Windows o Red Hat Linux que son sistemas operativos pagados.
- **Tipo de instancia de servidor:** el costo de paquetes computacionales depende de las necesidades que se requieren para ejecución de servicios, aplicaciones y sitios web sin ningún tipo de problemas. Parámetros claves para establecer los recursos asignados son CPU, memoria y almacenamiento en discos. Cada proveedor Cloud tiene su forma de administración y proceso de mejora de las máquinas y servicios virtuales, lo que significa que con el paso del tiempo los costos del tipo de instancia varían.
- **Ubicación:** el lugar de ubicación del centro de datos que se elige constituye el último aspecto clave que define los costos de servicios en la nube, este factor es de gran importancia ya que permite a las empresas realizar una optimización de los servicios que ofrecen a sus clientes. En la actualidad, AWS dispone de 61

centros de datos distribuidos por todo el mundo, Azure 54 y Google Cloud de 55 (Siranjeevi, 2019).

### **Amazon Web Services**

Con la tendencia de Pago por Uso (*PAYG*), los precios de *AWS* se ajustan en base a las necesidades computacionales del sistema a implementar, no se necesita adquirir soluciones de nube privada que son costosas y rápidamente se convierten en obsoletas. En *AWS* únicamente se requiere configurar la máquina virtual y la nube de Amazon Web Services se adapta a los requerimientos, por lo que únicamente se debe pagar por el tiempo de cómputo utilizado.

### **Microsoft Azure**

De la misma manera que *AWS*, la plataforma de Microsoft tiene muchas soluciones y precios, aunque los costos de uso son mucho más elevados. Con el propósito de liderar el mercado *Cloud*, Azure presenta un segmento para transportar aplicaciones hacia la nube de una manera más simple y rápida, pero a precios costosos. Dispone de una calculadora de gastos basadas en dos principios fundamentales: reducción de costos totales de la infraestructura local y establecer ahorros de costo estimados de acuerdo al peso de trabajo de las aplicaciones desarrolladas (Noriega, 2021).

### **Google Cloud Platform**

La plataforma de Cloud de Google es mucho más fácil de utilizar en comparación con las plataformas de los otros dos proveedores de servicios en la nube. Los precios de GCP también son accesibles y competitivos, Google estima que son un 35% más económicos con respecto a Azure y *AWS* al realizar cargas de cómputo y un 21% en tarifas de costos de almacenamiento. En la Tabla 21 se realiza una

comparativa de los precios según el tipo y características computacionales de la instancia.

**Tabla 21**

*Comparativa de precios de Cloud Computing*

Tipo	vCPU	Memoria [GiB]	RAM [GiB]	AWS [\$/hora]	Azure [\$/hora]	GCP [\$/hora]
<b>Propósito General</b>	2	8	4	0.0840	0.0928	0.0863
	4	16	8	0.1670	0.1856	0.1712
	8	32	16	0.3350	0.3712	0.3547
<b>Computación optimizada</b>	2	8	4	0.0850	0.0850	0.860
	4	16	8	0.1690	0.1700	0.1687
	8	32	16	0.3380	0.3400	0.3393
<b>Memoria optimizada</b>	2	8	4	0.1330	0.1330	0.1326
	4	16	8	0.2660	0.2660	0.2644
	8	32	16	0.5320	0.5320	0.5307

*Nota.* La tabla muestra los precios flexibles de las instancias on-demand. La categoría o modalidad On Demand también denominada como *PAYG* se centra en proporcionar recursos a los usuarios y cobrar por lo que se utiliza. Tomado de On Demand Azure vs AWS vs GCP pricing. Por Siranjeevi. R (2019)

### **Características**

Las características de los servidores Cloud Computing también constituye un factor importante para analizar, AWS dispone de diferentes algoritmos para separar sus productos y dividirlos por categorías. Azure también dispone de una amplia serie de prestaciones muy similares a AWS con el propósito de buscar que más clientes coloquen sus productos en su nube. Mientras que *Google Cloud Platform*, de manera progresiva continua su avance en el mundo del *Cloud Computing* con tres aspectos

clave para sus soluciones como: Infraestructura a prueba, almacenamiento potente con un análisis de datos serio y sin servidor, únicamente código (Noriega, 2021).

### ***Elección del servidor en la nube***

Basado en el análisis comparativo de costos presentado en la Tabla 21 y las características de servicios se elige a AWS como plataforma en la nube para el desarrollo de la aplicación web, configuración del *bróker* MQTT y almacenamiento de los datos transmitidos por los sensores, por su flexibilidad en cuanto a servicios, productos, características y múltiples opciones de pago dependiendo el tipo de instancia y parámetros computacionales. La instancia utilizada es de tipo t2. micro con 1 vCPU, 1 GiB de memoria, 30 GiB para almacenamiento, con desempeño de red de bajo a moderado y con compatibilidad para IPv6.

Azure y Google, basados en los parámetros anteriormente analizados también constituyen grandes proveedores Cloud, sin embargo, Azure orienta sus servicios hacia el sistema operativo Windows mientras que AWS dispone de instancias computacionales más competitivas con opciones gratis de múltiples sistemas operativos y con planes de almacenamiento similares a Azure. *Google Cloud* es una plataforma mucho más sencilla, nueva, flexible y presenta descuentos a largo plazo, no obstante, una vez comprado algún servicio no hay devolución del dinero invertido.

## Capítulo IV

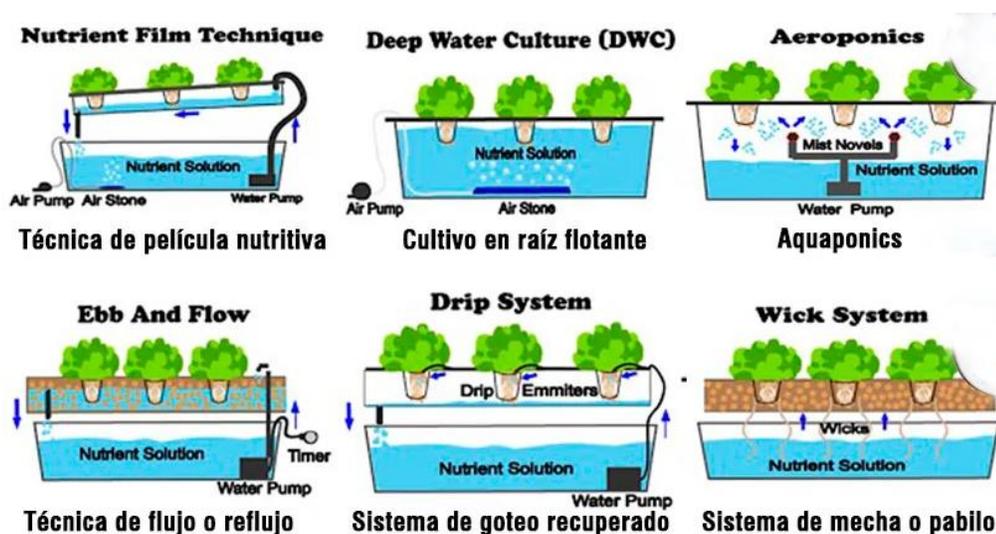
### Diseño e Implementación

#### Desarrollo del sistema hidropónico

Previo al diseño de la red de sensores inalámbricos se requiere desarrollar e implementar un sistema hidropónico. En la Figura 36, se observan muchos tipos de sistemas para hidroponía como: el sistema de goteo, sistema wick, aeroponía, NFT (del inglés, *Nutrient Film Technique*), sistema de Flujo y Reflujo (*Ebb and Flow*) y el sistema de cultivo de raíz flotante (Murcia & Chacón, 2018).

**Figura 36**

*Tipos de sistemas hidropónicos*



*Nota.* La figura presenta los diferentes sistemas para hidroponía con y sin el uso de sustrato. Tomado de Sistemas de hidroponía por Sánchez. E (2019).

El sistema NFT presenta características similares al sistema *Ebb* y *Flow* debido a que ambos hacen uso de bombas sumergibles en agua para distribuir la solución

nutritiva a las plantas, en la Tabla 22 se hace una comparación con las características de cada una de las técnicas para hidroponía.

**Tabla 22**

*Tipos de sistemas hidropónicos*

<b>Sistema hidropónico</b>	<b>Características</b>
<b>NFT</b>	Sistema más empleado a nivel mundial, utiliza la técnica de bombeo para hacer recircular la solución por la tubería para nutrir y oxigenar a las plantas. Ahorro de agua y solución nutritiva.
<b>Raíz Flotante</b>	Perfecto para plantas de tamaño pequeño (plantas aromáticas, culantro, etc.), simple y de bajo costo.
<b>Aeroponía</b>	Las raíces se encuentran colgadas en el aire, se hace uso de rociadores para nutrir a la planta. Sistema más complejo y de costo elevado
<b>Ebb y Flow</b>	Se realiza la inundación temporal de las bandejas mediante una bomba, requiere sustratos que absorban y luego la solución es drenada nuevamente al depósito.
<b>Goteo</b>	Funcionamiento parecido al riego por goteo, facilita la regulación del nivel de pH, mayor control del exceso de agua y los nutrientes para reutilizarlos.
<b>Wick</b>	No requiere de una bomba sumergible, simple y recomendado en plantas individuales con bajo consumo de agua

*Nota.* Los sistemas NFT, raíz flotante y aeroponía se diferencian de los otros sistemas porque no requieren del uso de sustratos, sus raíces se encuentran directamente en contacto con el agua. Tomado de Sistemas de Hidroponía por Sánchez. E (2019).

En base al análisis en la Tabla 22, se opta por la técnica NFT debido al ahorro importante de solución nutritiva y sobre todo agua en comparación con las otras técnicas, también porque permite monitorizar, controlar y acelerar el proceso de crecimiento de las plantas, reducir el espacio de implementación con una construcción e

instalación mucho más simple. La mayor desventaja que presenta este sistema es el riguroso cuidado del pH, nivel y estado de la solución nutritiva, sin embargo, al combinarlo con la red de sensores para monitorización en tiempo real se estima obtener excelentes resultados de cosecha.

Existen varios tipos de diseño para un sistema NFT, sin embargo, todos se basan en el uso de un contenedor con solución nutritiva la cual recircula a través de las tuberías. En la Figura 37, se observa el diagrama del sistema NFT, el cual está constituido de tuberías de material PVC, aunque también se pueden utilizar otros materiales como polietileno o poliuretano.

### **Figura 37**

*Diagrama del sistema hidropónico NFT*



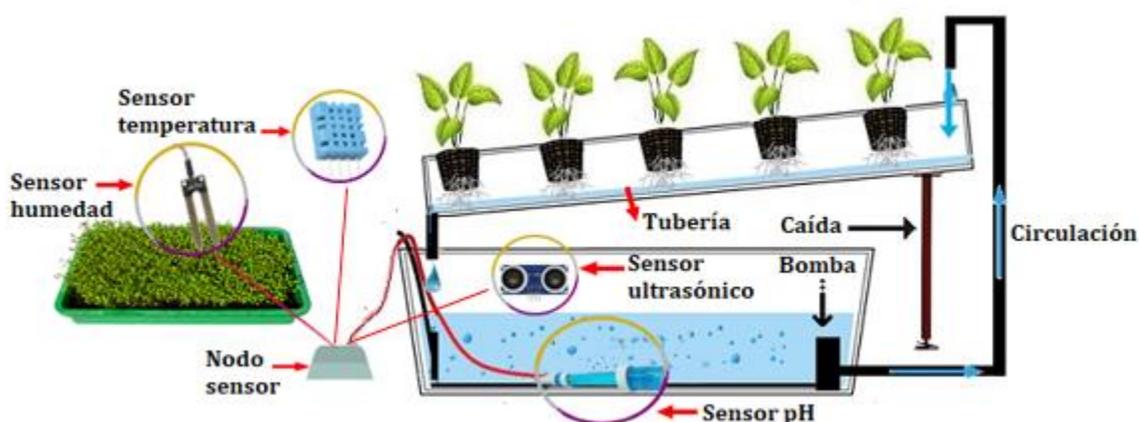
*Nota.* La imagen ilustra la arquitectura del sistema NFT compuesto de tuberías PVC, un contenedor para solución nutritiva y la estructura del invernadero. Adaptado de Hydroponic Culture System por Gómez. V (2020).

## Diseño del sistema hidropónico NFT

Para diseñar el sistema NFT, se utiliza los criterios y recomendaciones de (Mohammed, 2018), el cual menciona que los dos componentes fundamentales del sistema son: el canal de tubería de cultivo y el contenedor o depósito que almacena la solución nutritiva y agua. Los otros elementos son el semillero donde se realiza el proceso de germinación y la red de monitorización remota compuesto por los sensores de pH, humedad, temperatura y nivel. El diagrama del sistema completo a implementar se presenta en la Figura 38.

### Figura 38

*Sistema hidropónico NFT y red de sensores para monitorización remota*



*Nota.* La figura presenta el sistema hidropónico NFT, las raíces de las plantas al crecer forman una densa alfombra en el interior de los tubos, aspecto que debe ser tomado en consideración para evitar obstrucciones del flujo de solución. Para el sistema de retorno de solución nutritiva al contenedor se recomienda utilizar el método de goteo.

Como se observa en la Figura 38, para desarrollar un sistema hidropónico NFT se requiere los siguientes componentes:

1. Un contenedor para almacenar la solución nutritiva.

2. Una bomba para hacer circular el agua por todo el sistema.
3. Red de tubería para contener las plantas y distribuir la solución nutritiva.
4. Macetas de red para contener las plantas fijas en los orificios.
5. Sistema de retorno para dirigir la solución nutritiva de nuevo al contenedor.

En los orificios de la tubería se encuentran colocadas las macetas de red, también denominadas medios de cultivo, este elemento tiene la función de mantener firme a las plantas y permitir que las raíces absorban los nutrientes de la solución al entrar en contacto con el agua. Generalmente, como medios de cultivo se utilizan los materiales de la Tabla 2, aunque en el mercado agrícola se comercializa al cilindro de *foami* como macetas de red para producción a gran escala.

La bomba se encarga del proceso de suministro de agua por toda la tubería, adicionalmente se tiene un tubo de drenaje para retornar la solución nutritiva no absorbida por las raíces.

### ***Tuberías PVC***

El material del cual está elaborado cada ducto hidropónico es de plástico de Policloruro de Vinilo (PVC, del inglés *Polyvinyl chloride*), que cuenta con una protección de rayos ultravioletas, la inercia química de la composición de PVC y sus aditivos suscitan una resistencia al ataque de soluciones fertilizantes, aguas y suelos agresivos, así mismo, impide la proliferación de algas, hongos, moluscos, etc. (Plastigama, 2018).

En la Tabla 23 se presentan las características técnicas del tubo PVC provistas por el fabricante.

**Tabla 23***Características técnicas del tubo PVC*

<b>Propiedad</b>	<b>Unidad</b>	<b>PVC - 4</b>
<b>Densidad</b>	<i>gr/cm<sup>3</sup></i>	1.42
<b>Límite elástico mínimo</b>	<i>N/mm<sup>2</sup></i>	48
<b>Calor específico</b>	<i>Cal/gr °C</i>	0.25
<b>Resistividad a 20°C</b>	<i>Ohm/cm</i>	10 <sup>14</sup>
<b>Resistividad al impacto</b>	%	10
<b>Coefficiente de dilatación térmica</b>	<i>Mm/m °C</i>	0.05
<b>Conductividad térmica</b>	<i>kcal/m °C</i>	0.13
<b>Constante dieléctrica</b>	-	3
<b>Rugosidad hidráulica</b>	<i>n [Manning]</i>	0.008

*Nota.* El límite de elasticidad posibilita una elevada resistencia a presiones hidrostáticas y a esfuerzos generados por diversas cargas externas como, por ejemplo, sismos.

Tomado de Especificaciones para tuberías PVC, Plastigama (2018).

La tubería constituye el esqueleto del sistema, pues permite sostener de manera adecuada el peso de todo el cultivo y al mismo tiempo impedir el paso de rayos UV en el interior de la tubería que afectan a las raíces de las plantas. Según el fabricante, las tuberías vienen protegidas con aditivos de UV para evitar daños en el plástico al estar expuesto de manera directa con la luz del sol, Figura 39.

### Figura 39

*Tubería con protección UV*

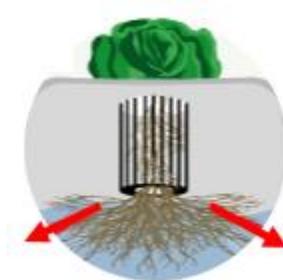


*Nota.* Adaptado de Plástico PVC con aditivos de protección UV por Hydro (2011).

El diseño circular de la tubería impide la obturación del flujo provocado por las raíces, facilita la circulación de la solución nutritiva y la oxigenación adecuada de las plantas, como se aprecia en la Figura 40.

### Figura 40

*Estructura circular de la tubería*



*Nota.* Adaptado de Diseño circular por Hydro (2011).

### **Criterio de longitud**

Según (Mohammed, 2018), la longitud de los tubos no debe superar los 9 metros o 30 pies, canales de tubería muy largos generan deficiencias en la distribución de los nutrientes, esto debido a que las plantas colocadas en los primeros orificios retienen la

mayor cantidad de nutrientes y provocan que las plantas ubicadas en los últimos boquetes del sistema tengan un pésimo desarrollo. Por eso, en canales largos se recomienda añadir una fuente extra de solución nutritiva en la mitad del tubo o canal de cultivo, las características del sistema de tubería desarrollado se presentan en la Tabla 24.

**Tabla 24**

Características de la tubería

<b>Parámetro</b>	<b>Característica</b>
<b>Coloración</b>	Blanco
<b>Longitud del tubo</b>	3 metros
<b>Número de perforaciones por tubo</b>	10
<b>Diámetro nominal del tubo</b>	4 pulgadas (100mm)
<b>Diámetro exterior del tubo</b>	11.43 cm
<b>Diámetro interior del tubo</b>	10.83 cm
<b>Grosor</b>	3mm
<b>Peso aproximado</b>	1.61 kg/m

*Nota.* Tomado de Especificaciones para tuberías PVC, Plastigama (2018).

### ***Selección del contenedor***

Uno de los factores cruciales para el crecimiento exitoso de las plantas en el sistema hidropónico, es la selección del tamaño del contenedor. Si se elige un contenedor muy pequeño las plantas no recibirán la cantidad de agua y nutrientes apropiados. El volumen del tanque tiene una relación directa con el número y el tipo de plantas a cultivar, por ejemplo, para el cultivo de pepino en verano y en fase productiva,

el consumo de solución nutritiva es de 3 litros por planta al día (Carrasco & Izquierdo, 1996).

**Tabla 25**

*Parámetros de cálculo para la capacidad del contenedor*

<b>Especie</b>	<b>Volumen de solución (L planta/día)</b>	<b>Densidad (planta/m<sup>2</sup>)</b>	<b>Capacidad requerida del tanque (L/m<sup>2</sup>)</b>
<b>Lechuga</b>	0.3	24	9
<b>Pepino</b>	3	5	19

*Nota.* La tabla muestra el consumo estimado de una planta en su máximo estado de desarrollo, se debe tomar en consideración un 20 o 25% más del volumen consumido para excedente. Tomado de Cálculo de la capacidad del estanque según la especie cultivada, Carrasco. G & Izquierdo. J (1996, p.17).

El sistema hidropónico en análisis se considera para una producción experimental de 40 plantas (6 de tomate y 34 de lechuga), donde se utilizan los criterios presentados en la Tabla 25 para determinar la capacidad requerida del tanque, como se presenta a continuación:

$$C_{req} = \frac{\text{numero de plantas} \times C. \text{requerida}}{\text{densidad}}$$

$$C_{req} = \frac{34 \times 9}{24} + \frac{6 \times 19}{5}$$

$$C_{req} = 35.6 L$$

En contenedores pequeños, el volumen de solución disminuye de manera rápida por lo que se requiere que el nivel no sea menor al 70% (Mohammed, 2018). Una vez determinada la necesidad máxima de agua para todo el sistema NFT, se selecciona un contenedor que cumpla con las características de volumen requeridas.

$$C_{total} = 35.6 L + 20\% \times C_{req} \approx 45L$$

**Tabla 26***Dimensiones del contenedor*

<b>Parámetro</b>	<b>Valor</b>
<b>Volumen bruto</b>	50 L
<b>Profundidad</b>	45 cm
<b>Ancho</b>	50 cm
<b>Alto</b>	53 cm
<b>Peso bruto</b>	2.8 kg

*Nota.* La tabla expone las dimensiones del contenedor, el material del cual está elaborado es de polietileno lo que lo hace liviano y resistente.

El volumen del contenedor debe ser de mínimo 50 L que permita disponer del 20% extra de la capacidad requerida, es decir una reserva de 7.2 L de solución. Las dimensiones del tanque se visualizan en la Tabla 26, en la investigación de (Resh, 2004) se recomienda no llenar a tope el tanque, sino entre un 80% y 90% para evitar desbordes de solución.

### **Selección de la bomba**

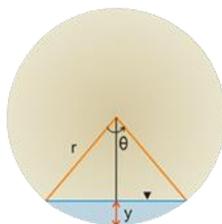
Para poder determinar la potencia requerida para el sistema, se considera el caudal y la distancia recorrida. Por medio de la ecuación (9) de Hazem-Williams es posible definir el caudal, para ello se requiere calcular previamente el área de flujo ( $A$ ), perímetro mojado ( $P$ ), radio hidráulico ( $R_H$ ) y el ángulo ( $\theta$ ) con los parámetros de la Tabla 27.

**Tabla 27***Parámetros para el cálculo de caudal*

Parámetro	Valor
<b>Diámetro interior</b>	10.83 cm
<b>Profundidad de flujo</b>	1 cm
<b>Pendiente de fondo</b>	1%
<b>Coefficiente</b>	150

*Nota.* La tabla presenta los parámetros para determinar el caudal provistos por el fabricante. La pendiente de fondo se define como la relación entre la longitud del tubo y la altura a la cual se coloca, cuanto más grande es el valor de la pendiente de fondo mayor es el caudal.

La Figura 41 representa el radio interno del tubo ( $r$ ), profundidad de flujo ( $y$ ), pendiente de fondo ( $S$ ) y el coeficiente ( $C$ ) que está sujeto a la rugosidad del tubo. Cabe mencionar que el valor típico de  $C$  empleado para el diseño con tuberías de policloruro de vinilo es de 150.

**Figura 41***Tubo circular*

*Nota.* El coeficiente  $C$  depende de la edad de la tubería, con el paso del tiempo tiende a volverse más rugosa y el coeficiente disminuye.

$$Q = 0.2787 \times C \times A \times R_H^{2/3} \times S^{1/2} \quad (9)$$

$$\theta = 2 \times \cos^{-1} \left[ 1 - \frac{y}{r} \right] \quad (10)$$

$$A = 0.5 \times r^2 \times (\theta - \sin\theta) \quad (11)$$

$$P = r \times \theta \quad (12)$$

$$R_H = \frac{A}{P} \quad (13)$$

En base a las ecuaciones anteriormente planteadas se determina el caudal.

$$\theta = 2 \times \cos^{-1}[1 - 0.055] = 1.21 \text{ rad}$$

$$A = 0.5 \times 0.054^2 \times (1.21 - \sin(1.21)) = 4 \times 10^{-4} \text{ m}^2$$

$$P = 0.054 \times 1.21 = 0.06 \text{ m}$$

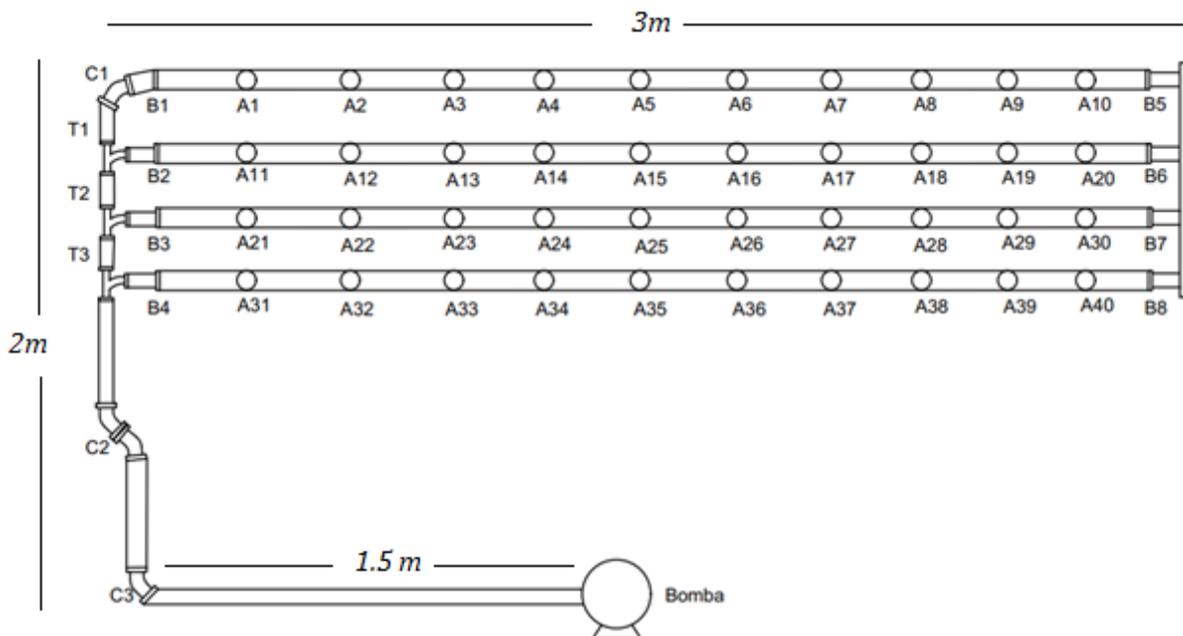
$$R_H = \frac{0.0029 \text{ m}^2}{0.12 \text{ m}} = 6 \times 10^{-3} \text{ m}$$

$$Q = 5 \times 10^{-5} \frac{\text{m}^3}{\text{s}} = 180 \frac{\text{L}}{\text{h}}$$

Posteriormente, para definir la potencia de la bomba se define el caudal de trabajo que depende del total de tubos.

$$Q_{\text{trabajo}} = 180 \frac{\text{L}}{\text{h}} \times 4 = 720 \frac{\text{L}}{\text{h}} = 3.16 \frac{\text{gal}}{\text{min}}$$

Después de calcular el caudal de trabajo se procede a establecer la distancia total que recorre el agua por todo el sistema en base a la distribución de la Figura 42.

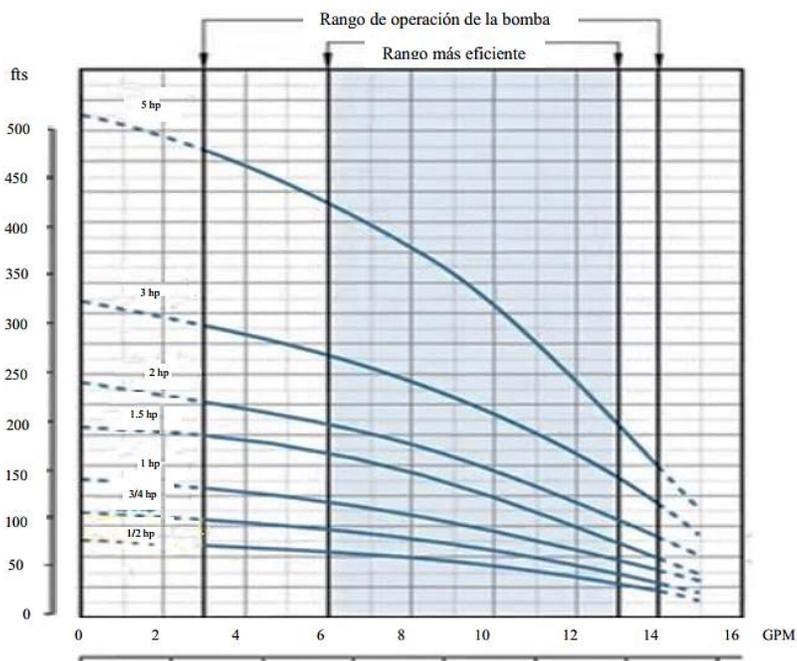
**Figura 42***Diseño del sistema hidropónico*

*Nota.* Desarrollado en el *software Autocad.*

La nomenclatura del diseño establece que desde A1 hasta A40 son los orificios de cultivo, B1 a B8 son los tapones de tubería, C1 a C3 son los codos y T1 a T3 son las Tee de acero utilizadas. En base a este diagrama, se determina que el recorrido total de agua por el sistema hidropónico es de aproximadamente 52.5 ft o 16 m. Una vez definida la distancia recorrida en pies, se selecciona el tipo de bomba en base a la Figura 43, que está en función de los pies y galones por minuto.

### Figura 43

#### Selección de bomba sumergible



*Nota.* Tomado de gráfica de selección de bomba sumergible, por Murcia. J & Chacón.

L. (2018, p.48)

Finalmente, se establece que para 52.5 ft de recorrido y 3.16 galones por minuto, se requiere una bomba sumergible de 1/2 hp. La bomba de la Figura 44, será la encargada de realizar el bombeo de agua con solución nutritiva por todo el sistema.

**Figura 44***Bomba sumergible*

*Nota.* Bomba sumergible de 1/2 hp.

En la Tabla 28, se observan las especificaciones técnicas de la bomba sumergible.

**Tabla 28***Parámetros técnicos bomba sumergible*

<b>Parámetro</b>	<b>Característica</b>
<b>Material</b>	Plástico y acero inoxidable
<b>Potencia</b>	1/2 Hp
<b>Consumo</b>	2.8 A
<b>Frecuencia</b>	60 Hz
<b>Profundidad máxima</b>	8m
<b>Velocidad</b>	2.850 rpm
<b>Peso</b>	3.7 kg
<b>Altura máxima</b>	10 m

*Nota.* Tomado de Especificaciones de la bomba sumergible, por Truper (2021).

### ***Solución nutritiva y plantación***

Los elementos minerales nutritivos que las plantas necesitan son proporcionados de manera exclusiva por los componentes de la Tabla 1, que componen la solución nutritiva y que son disueltos en el agua. La constante circulación de la solución en el sistema, hace que se requiera de una adición efectiva de nutrientes conforme éstos son absorbidos por las raíces. De este modo, se recurre al uso soluciones concentradas que son suministrados en pequeñas cantidades al agua. Por lo general, la formulación concentrada viene separada en dos soluciones, conocidas como Solución Concentrada A y Solución Concentrada B, compuestas de los elementos presentados en la Tabla 29.

**Tabla 29**

*Componentes de Soluciones Concentradas*

<b>Solución Concentrada</b>	<b>Componentes</b>
<b>A</b>	Nitrato de Calcio, Nitrato de Potasio
<b>B</b>	Sulfato de Magnesio, Sulfato de Cobre, Sulfato de Manganeso, Sulfato de potasio, Ácido Bórico, Molibdato de Amonio, Quelato de Hierro

*Nota.* La tabla presenta los fertilizantes de las soluciones concentradas, el objetivo de separar los componentes de las soluciones es evitar que se produzcan reacciones de ciertas sales Carrasco. G & Izquierdo. J (1996, p.33).

### **Siembra y trasplante**

La siembra de semilla de realizarse de manera adecuada y en base a los requerimientos de cada planta, en la Figura 45 se observa las partes de la mano involucradas en el proceso de siembra.

**Figura 45***Siembra de semillas*

*Nota.* Tomado de Zonas de contacto durante la siembra, por Quirós. D (2011, p.62).

En condiciones óptimas el periodo de germinación dura de 4 a 6 días en lechugas y de 7 a 10 días en tomates, cuando la plántula dispone de 6 hojas se realiza el proceso de trasplante a la maceta de cultivo. En la Figura 46, se observan las partes de la mano que deben ocuparse para el proceso de trasplante.

**Figura 46***Trasplante de semillas*

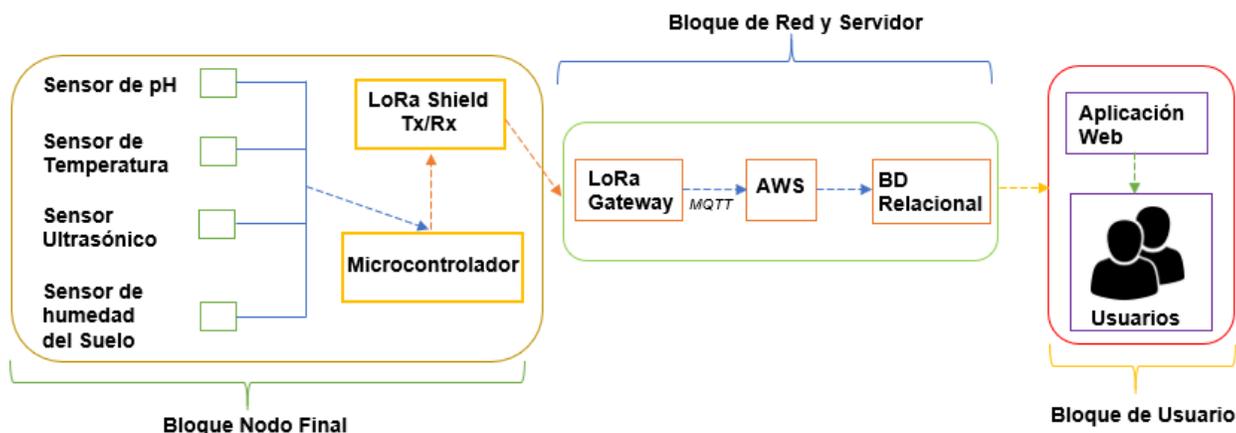
*Nota.* Tomado de Zonas de contacto durante el trasplante, por Quirós. D (2011, p.63).

## Arquitectura del sistema de monitorización

En base a los elementos y objetivos del proyecto se diseña la estructura del sistema de monitorización. En el diagrama de la Figura 47, se observa desde una perspectiva general la arquitectura compuesta de tres etapas descritas a continuación.

**Figura 47**

*Diagrama de bloques del sistema de monitorización*



*Nota.* La figura muestra los bloques o etapas que conforman la arquitectura del sistema de monitorización.

- Bloque Nodo Final: constituido por un Arduino uno al cual se encuentran conectados cuatro sensores: temperatura, humedad del suelo, pH y ultrasónico, los cuales se encargan de sensar las variables físicas en el cultivo hidropónico y LoRa *Shield* que procesa y transmite los datos sensados hacia el *Gateway* de LoRa mediante la frecuencia de 915 MHz.
- Bloque Servidor y Red: constituye la infraestructura de la red, configuración del servidor de AWS, *Gateway* LoRa, *bróker* MQTT, protocolos y puertos de red. Tiene la función de recibir los datos transmitidos por el nodo final y decodificar la carga útil en tiempo real con

el uso de la tecnología Node JS para almacenar la información en una base de datos de tipo relacional y mostrarlos en la plataforma Web IoT.

- Bloque de Usuario: integra a la plataforma IoT, la cual mediante tecnologías como *JavaScript*, *PHP* y *SQL* procesa los datos para visualización y análisis. En la plataforma los usuarios tienen la capacidad de establecer *Set Points* y recopilar la información histórica de monitorización.

### **Diseño del Nodo final**

El nodo final o nodo sensor del sistema de monitorización está compuesto de un Arduino uno como microcontrolador, los cuatros sensores anteriormente definidos en la Figura 48, el transceptor LoRa *Shield*, una fuente de alimentación y un *Buck* convertidor DC-DC reductor. El microcontrolador Arduino constituye la parte central del sistema, pues se encarga de controlar los cuatro sensores que conforman la red y que permiten monitorizar parámetros primordiales como pH del agua, nivel del contenedor de la solución, temperatura ambiental y humedad del suelo en los semilleros. Además, para el proceso de automatización de la bomba del sistema NFT se requiere de un relé.

### **Diagrama de Conexiones**

El transceptor LoRa Shield se conecta por encima del Arduino y se encarga de realizar el proceso de comunicación entre el *Gateway* LoRa y la red de sensores. Previo al diseño del diagrama de conexión de los dispositivos, se tiene en consideración la distribución de los pines de la *Shield* provista por el fabricante y definida en la Tabla 30.

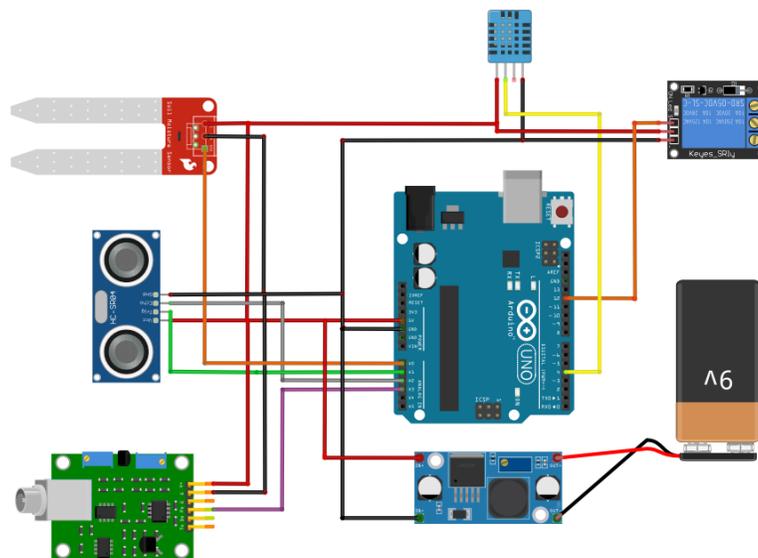
**Tabla 30**

*Distribución de los pines de LoRa Shield*

<b>Pines de LoRa Shield</b>	<b>Disposición</b>
<b>GND</b>	Tierra
<b>A0 - A5</b>	Pines analógicos utilizables
<b>D0 - D1</b>	Pines digitales utilizables
<b>D2</b>	Pin digital no utilizable
<b>D3 - D5</b>	Pines digitales utilizables
<b>D6 - D8</b>	Pines digitales utilizables bajo condiciones
<b>D9</b>	Pin digital no utilizable
<b>D10 - D13</b>	Pines digitales utilizables bajo condiciones
<b>SLC, SDA, REF</b>	Pines utilizables

*Nota.* La tabla presenta la distribución de pines de la placa LoRa Shield y su disponibilidad. Tomado de LG01N/OLG01N LoRa Shield (p.5) por Dragino, 2020.

Para alimentación del Nodo Final se dispone de una fuente que provee 9 y 12 voltios, razón por la cual entre la fuente de alimentación y Arduino se utiliza un convertidor de tipo DC-DC reductor, cuya función es proveer 5 voltios estables para que el microcontrolador no experimente algún tipo de daño generado por caídas o picos de tensión. Adicionalmente se recomienda del uso de un disipador pequeño que debe ser colocado de manera directa sobre el regulador, con el propósito de reducir el calor generado. Finalmente, los sensores, el relé y el convertidor *step-down* se conectan al microcontrolador en base al diagrama esquemático de la Figura 48.

**Figura 48***Diagrama físico del Nodo Final*

*Nota.* La figura indica el diagrama de conexión de los componentes electrónicos que conforman al Nodo Final

### ***Configuración del software de la red de sensores***

Previo a la adquisición de los datos de monitorización se requiere configurar al microcontrolador, para ello es necesario añadir algunas librerías al entorno de desarrollo de Arduino "IDE", con la intención de configurar mediante programación el funcionamiento de cada uno de los sensores que conforma la red de monitorización y así mismo para el envío y recepción de datos con el transceptor LoRa.

#### **Componentes y librerías**

Antes de realizar la configuración del microcontrolador se debe instalar el driver CH340, controlador que permite el manejo y programación de circuitos integrados desde la interfaz USB. Las librerías y componentes requeridos para el desarrollo del software del Nodo Final son:

Tabla 31

Componentes del software del Nodo Final

Categoría	Nombre	Descripción
<b>Librería</b>	LoRa.h	Permite la transmisión y recepción de datos con <i>LoRa</i>
	lmic.h	Permite la transmisión y recepción de datos con <i>LoRa</i> y hacer uso de la red <i>LoRaWAN</i>
	SPI.h	Realiza el proceso de comunicación entre <i>LoRa</i> Shield y la placa de Arduino
	dht.h	Obtiene el valor en tiempo real de la humedad y temperatura ambiente.
	SimpleTimer.h	Librería utilizada para trabajar con el tiempo
<b>Funciones</b>	DHT.temperature ()	Permite obtener el valor de la temperatura ambiental en un rango de 0° a 60°
	DHT.humidity ()	Obtiene el valor de la humedad relativa ambiental en un rango de 0 a 100.
	void mediciones ()	Función para concatenar los datos de medición de los cuatro sensores y formar el payload.
	void setup ()	Función para inicializar, sincronizar y verificar el proceso de comunicación <i>LoRa</i>
	void valorPH ()	Permite obtener las mediciones del sensor de <i>pH</i>
	SendData ()	Función que contiene métodos para el envío de la carga útil
	beginPacket ()	Permite iniciar el envío del payload con el uso de <i>LoRa</i>
	endPacket ()	Finaliza el envío de la carga útil
	int humedadSuelo ()	Permite obtener el porcentaje de humedad del suelo
	float ping ()	Permite medir el tiempo entre pulsos en <i>us</i> para luego convertir a cm y obtener el nivel del agua.

<b>Variables</b>	node_id	Id del nodo final para identificación dentro de la red <i>MQTT</i>
	counterTX	Contador del número de transmisiones correctas
	DHT11_PIN 3, sensorH A0, EchoPin A2, TriggerPin A1, bomba_pin 13, analog_PH	Variables de asignación de los pines de ocupación.
	int TriggerPin, EchoPin	Variables de asignación del pulso y medición del tiempo entre pulsos.
	float duracion, distanciaCm;	Variables de almacenamiento de la distancia en cm y el tiempo en <i>us</i>
	int valorHumedad	Variable para el mapeo de la lectura del sensor de humedad
	float temperatura, humedad	Variables de almacenamiento de la temperatura y humedad ambiental
	float tanque, totalNivel	Variable de definición de la altura del tanque en cm y el nivel de agua en porcentaje.
	float phVol, phVal	Variables que definen el valor de tensión y almacenamiento de la medición del sensor de <i>pH</i> .
	uint8_t datasend	Variable entera sin signo que contiene la carga útil

*Nota.* La tabla sintetiza de manera general las librerías, funciones y variables utilizadas en la programación del microcontrolador.

### **Calibración de los sensores**

Previo a la implementación de la red de monitorización de las variables física de humedad, pH, nivel y temperatura en el sistema NFT, se requiere verificar que los sensores se encuentren adecuadamente calibrados y que las mediciones sean precisas. Para ello se utilizan equipos patrones para comparar los datos sensados en un tiempo y lugar específico. El proceso de medición se realiza por el lapso de dos horas, de manera que se obtiene un total de 120 muestras.

## Figura 49

### *Equipos patrones*



*Nota.* La figura presenta los equipos patrones utilizados en la verificación de medición de variables.

En la Figura 49, se observa los equipos utilizados para el proceso de comparación de medidas, los cuales son descritos a continuación:

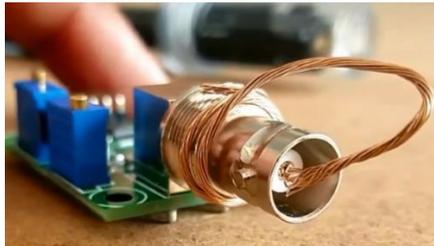
- Multímetro digital *Mastech*: dispositivo digital para medir la temperatura ambiente.
- Medidor digital de pH *Pancellent*: equipo digital para determinar el nivel de pH en soluciones líquidas
- Módulo de humedad del suelo FC-28: dispositivo medidor de la humedad del suelo, incorporado con una tarjeta de acondicionamiento YL-38.
- Medidor *Yegren*: dispositivo medidor de pH y humedad del suelo, temperatura e intensidad de la luz.
- Sensor ultrasónico hc-sr04: sensor encargado de medir el nivel de solución en el tanque.

### **Calibración del sensor de pH**

Para calibrar el sensor de pH se requiere de un cable conductor para cortocircuitar el centro y la parte externa del conector BNC de la placa pH-402C como se presenta en la Figura 50.

#### **Figura 50**

*Proceso de calibración del sensor de pH*



*Nota.* La figura muestra el proceso de calibración para la sonda de pH.

Una vez cortocircuitado el conector BNC se realiza la alimentación de la placa con los 5 voltios de Arduino lo que produce una salida de 2.5 V en el pin Po analógico. Se utiliza un multímetro para comprobar que se tiene los 2.50 voltios en la salida, con un valor diferente se debe ajustar el potenciómetro de Offset hasta conseguir este valor de voltaje. Por medio de la calibración se determina que 2.5 voltios equivale a un pH neutro de 7.

#### **Desarrollo del software**

Luego de utilizar las soluciones con niveles de pH de 4 y 6.86, se obtiene en el pin Po los valores de tensión de 3.04 y 2.54 V respectivamente. La sonda de pH es un sensor de tipo lineal, por lo que se usa los valores de pH y voltaje obtenidos para establecer una ecuación que permita convertir la tensión a pH, para ello se utiliza la ecuación de la recta  $y = mx + b$ . Al eje  $y$  se asigna el nivel de pH y al eje  $x$  la tensión,

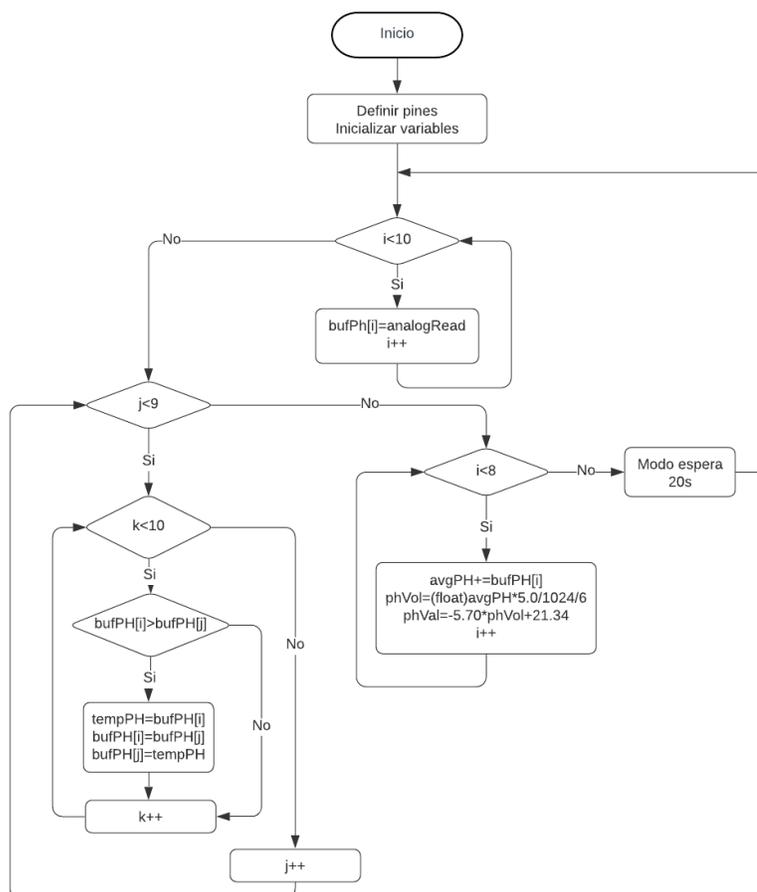
por lo que además se necesita calcular el valor de  $m$  y  $b$ . La ecuación (14) define el nivel de  $pH$  en función de la tensión.

$$y = -5.70x + 21.34. \quad (14)$$

Mediante el diagrama de flujo de la Figura 51 se realiza la programación. Se requiere obtener por lo menos 10 muestras en la entrada analógica A3, ordenarlas y descartar la medición más alta y más baja. Luego se procede a calcular el promedio únicamente entre las seis muestras restantes, se convierte el valor de la media en tensión y se almacena en la variable  $phVol$ . Posteriormente con la ecuación (14) se convierte de tensión a  $pH$ .

**Figura 51**

*Diagrama de flujo para configuración del sensor de pH*

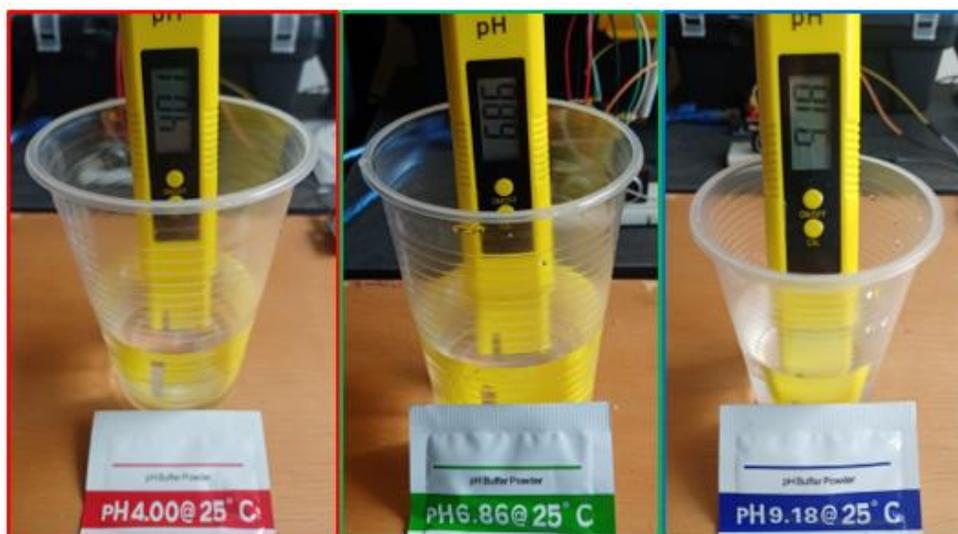


### Respuesta del sensor de pH

Una vez calibrado la sonda se realiza mediciones tanto con el equipo patrón como con el sensor. Se utiliza 4 soluciones para comprobar la precisión de las medidas: la primera solución tiene un pH de 4, la segunda un pH de 6.86 y la tercera un pH de 9.18. Previo a las mediciones con el sensor se utiliza el medidor digital de pH y se obtiene los resultados de la Figura 52.

### Figura 52

*Mediciones de pH con equipo patrón*



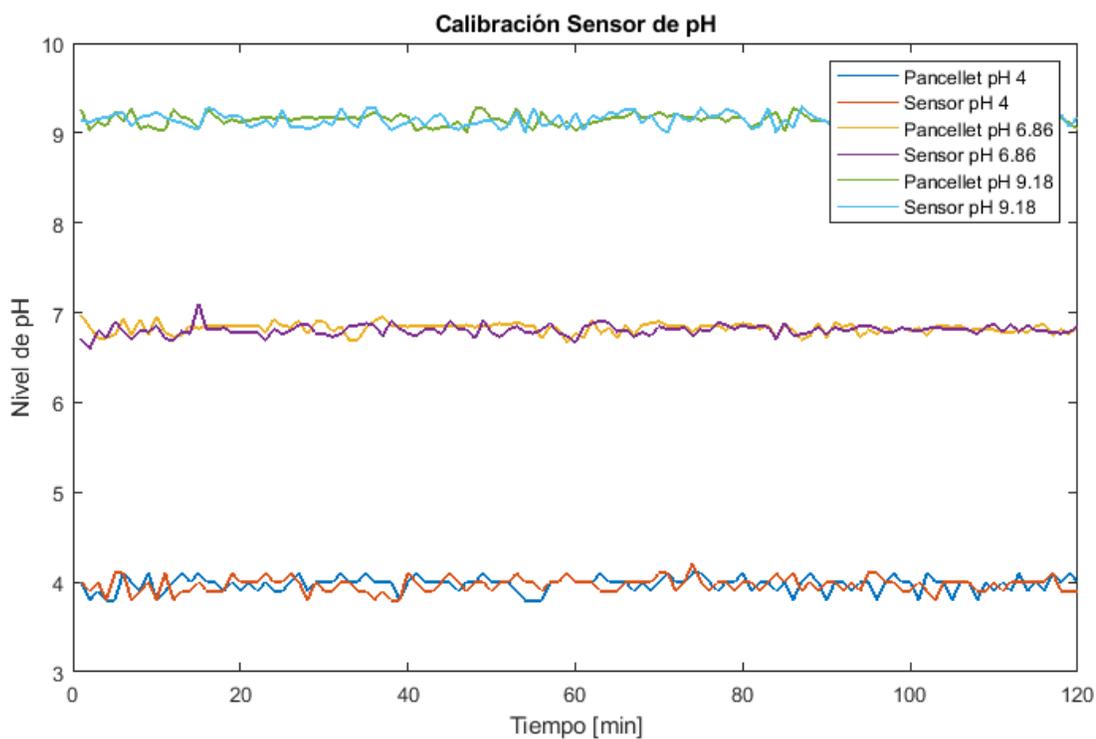
*Nota.* Las mediciones de pH se realizan en la escala de 0 a 14 para determinar el nivel de acidez y alcalinidad de una sustancia, un valor de 0 establece que la sustancia es muy ácida, 14 muy alcalina y 7 neutra.

Los resultados de las curvas de medición se observan en la Figura 53, el comportamiento del sensor de pH a lo largo del tiempo se ajusta a la curva obtenida por el equipo patrón, sin embargo, se observa que el dispositivo Pancellent ofrece una

mayor cantidad de fluctuaciones en las medidas debido a su rápida respuesta. (Ramirez & Ruales, 2021)

### Figura 53

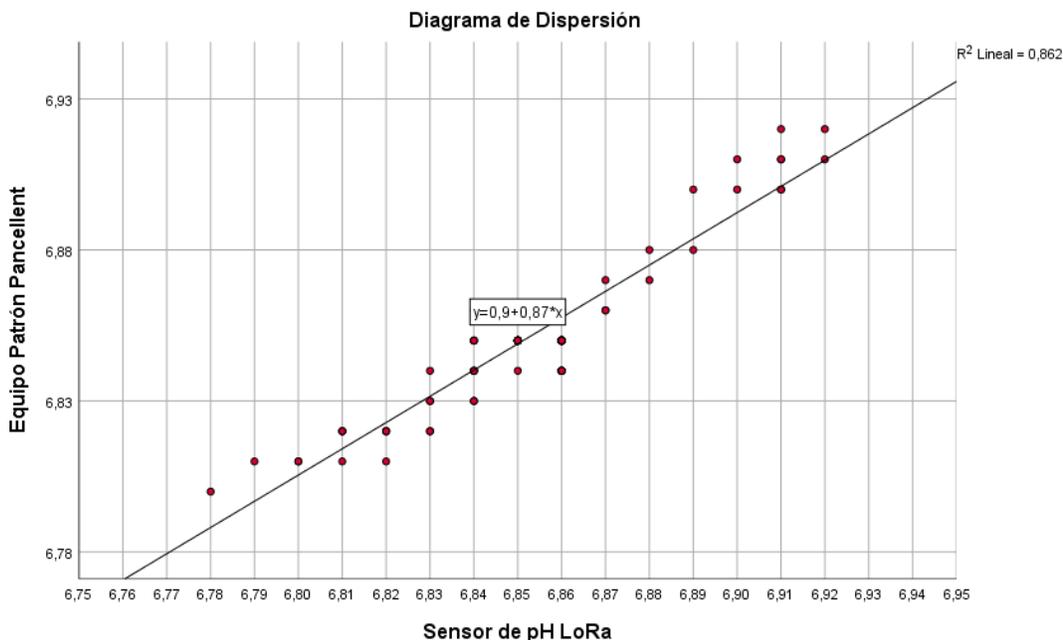
*Curva de calibración sensor de pH*



Con los datos obtenidos por el equipo patrón y el sensor se realiza el diagrama de dispersión, la finalidad de este análisis es determinar si las mediciones realizadas por ambos equipos se encuentran relacionadas. Mediante la Figura 54, se establece que las mediciones presentan una tendencia idéntica, el coeficiente de correlación lineal tiene un valor de 0.862 muy cercano a 1 lo que exhibe un tipo de correlación fuertemente positiva.

**Figura 54**

*Diagrama de dispersión mediciones de pH*



### ***Configuración del sensor de temperatura***

El sensor de temperatura DHT11 debe ser conectado en la entrada D3 de la Shield *LoRa*, debido a que la señal digital formada por los datos de humedad, temperatura y checksum requieren ser procesados por el microcontrolador. De las dos mediciones realizadas por el sensor únicamente para el proyecto se utilizan los datos de temperatura. La comunicación entre el microcontrolador y el sensor empieza con una señal de inicio enviada por Arduino, la cual es respondida por el sensor con un cambio de modo de espera a ejecución. Al terminar la señal de envío, el sensor remite un paquete de datos de 40 bits formados por los 16 bits de humedad, 16 de temperatura y los 8 bits restantes de checksum. Una vez terminado el proceso de envío de información, el procesador del sensor retorna a su estado de espera.

### Desarrollo del software

Para realizar el proceso de medición de temperatura y humedad ambiental, en el respectivo *sketch* de configuración se requiere agregar la librería `dht.h`, luego establecer el pin de entrada digital al cual se conecta la salida del sensor, definir el tipo de sensor utilizado e inicializar las características en una variable. El código de configuración es el siguiente:

```
1 #include <dht.h>
2 #define DHT11_PIN 3
3 #define DHTTYPE DHT11
4 DHT dht11 (DHTPIN, DHTTYPE) ;
```

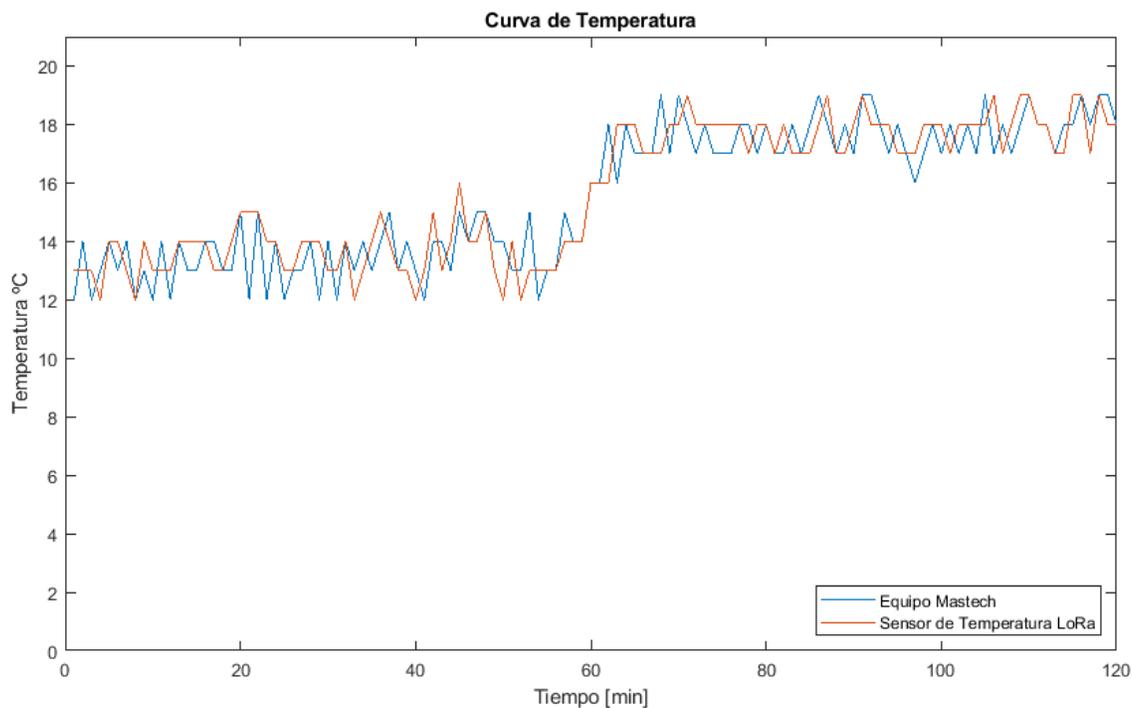
La variable `dht11` permite el proceso de comunicación entre el sensor y el microcontrolador

### Respuesta del sensor de Temperatura

Se realiza mediciones tanto con el equipo patrón como con el sensor configurado en el *sketch* de Arduino. Las pruebas se efectúan al interior del invernadero donde se encuentra instalado el sistema hidropónico, los resultados de las curvas de medición se observan en la Figura 55, el comportamiento del sensor de temperatura a lo largo del tiempo se ajusta a la curva obtenida por el equipo patrón, sin embargo, se observa que el dispositivo Mastech ofrece una mayor cantidad de fluctuaciones en las medidas debido a su rápida respuesta (Ramirez & Ruales, 2021)

**Figura 55**

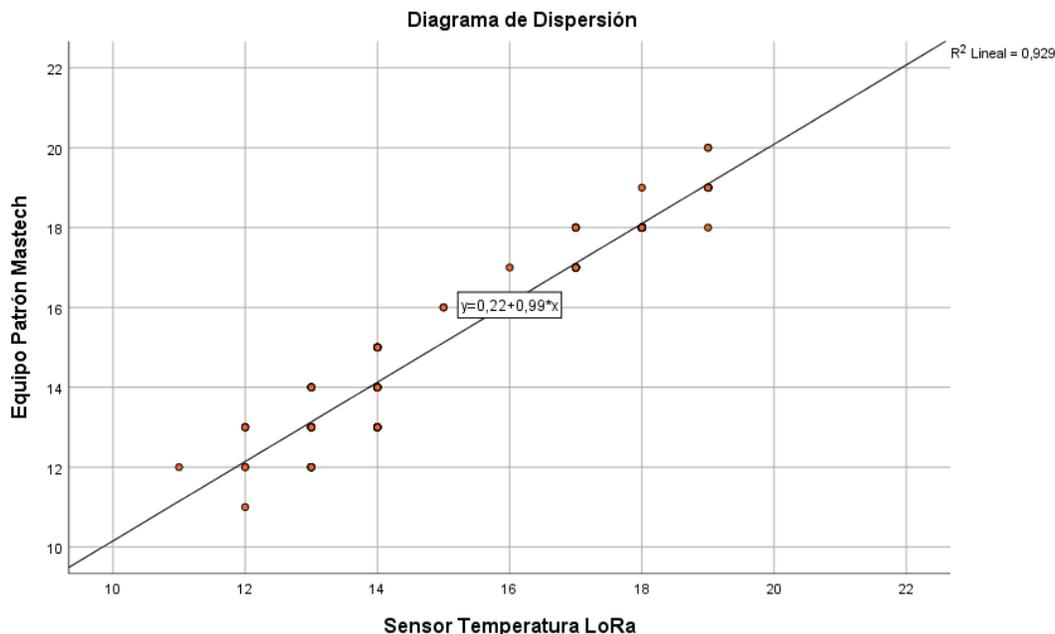
Curva de calibración sensor de temperatura



Con los datos obtenidos por el equipo patrón y el sensor se realiza el diagrama de dispersión, la finalidad de este análisis es determinar si las mediciones realizadas por ambos equipos se encuentran relacionadas. Mediante la gráfica de la Figura 56, se establece que las mediciones presentan una tendencia idéntica, el coeficiente de correlación lineal tiene un valor de 0.929 muy cercano a 1 lo que exhibe un tipo de correlación fuertemente positiva.

**Figura 56**

*Diagrama de dispersión mediciones de temperatura*



### ***Configuración del sensor de humedad del suelo***

El sensor de humedad no requiere un proceso de calibración, su configuración es realizada en base a la programación en el IDE de Arduino, para realizar las mediciones basta con conectar el sensor a cualquier entrada analógica del microcontrolador. Con la finalidad de disponer de mediciones de humedad en un rango de 0 a 100% en vista de que la escala original del sensor es de 0 a 1023, en la programación se debe efectuar una división del valor proporcionado por el sensor y 10 o utilizar la función map para hacer un mapeo de escala.

### **Desarrollo del software**

Para realizar el proceso de medición de humedad del suelo, en el respectivo sketch de configuración se requiere establecer el pin de entrada analógica, luego se inicializa una variable que almacena las lecturas realizadas por el sensor, después

mediante la función map se hace un cambio de escala de 0 a 100, por defecto el sensor maneja la escala de 0 a 1023. Un valor de 0 equivale a un 100%, es decir totalmente húmedo y 1023 un suelo absolutamente seco. El código de configuración se presenta a continuación:

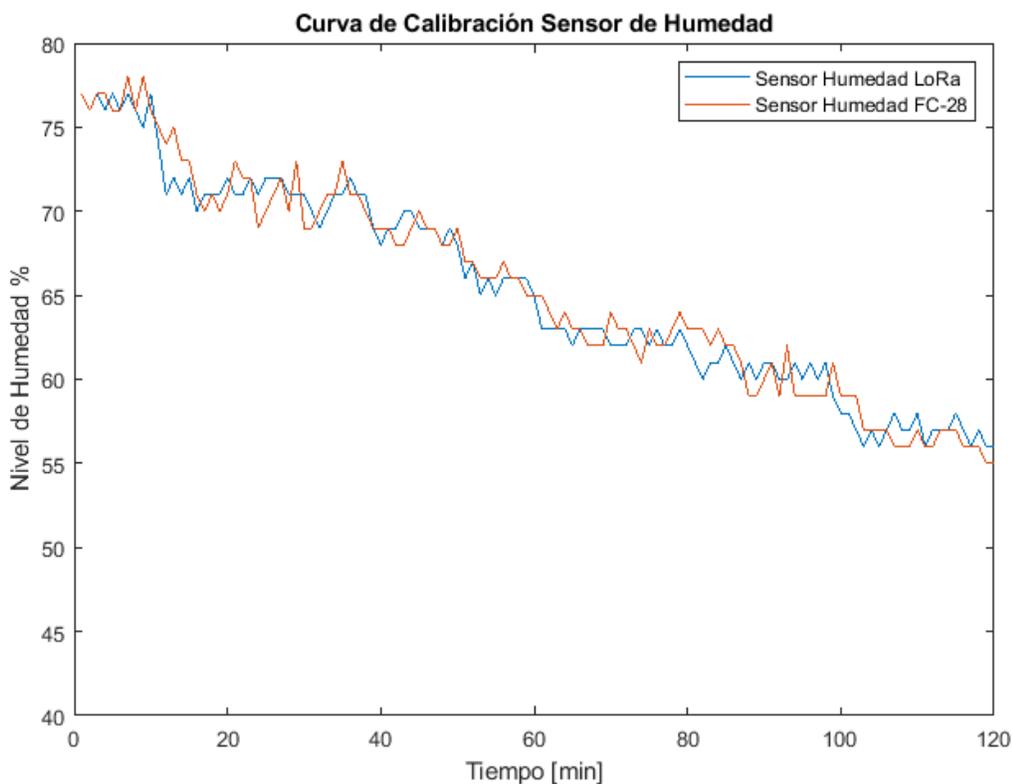
```
1 #define sensorH A0
2 int valorHumedad=0;
3 int valorHumedad=map(analogRead(sensorH), 0, 1023, 100, 0);
```

### **Respuesta del sensor de humedad**

Se realiza mediciones tanto con el equipo patrón como con el sensor. Las pruebas se efectúan en los semilleros de germinación, los resultados de las curvas de medición se observan en la Figura 57, el comportamiento del sensor de humedad a lo largo del tiempo se ajusta a la curva obtenida por el otro sensor utilizado para la verificación de la calidad de los datos medidos.

**Figura 57**

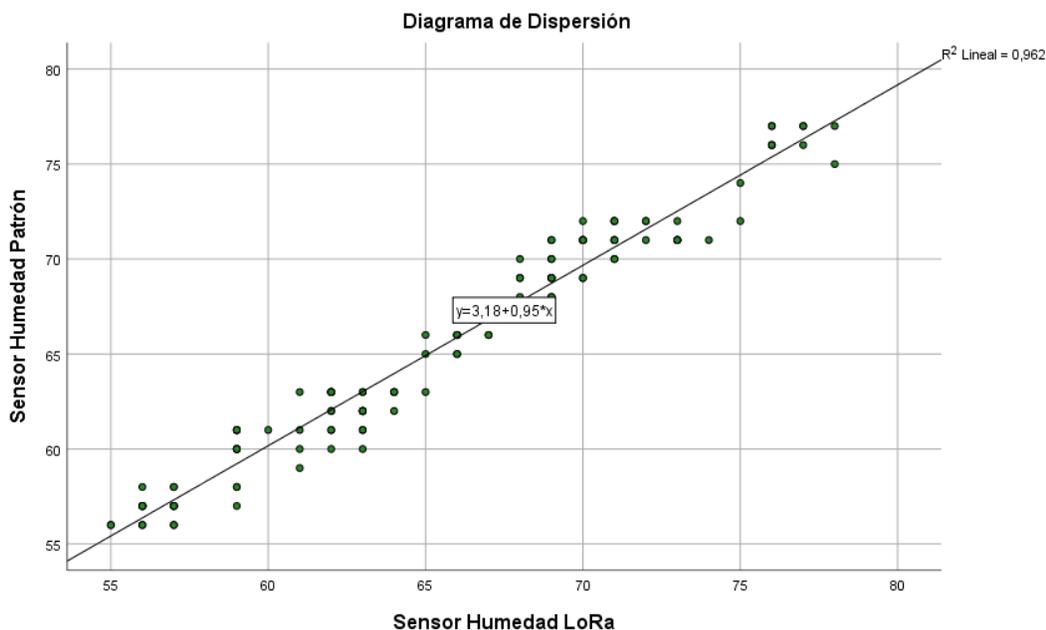
*Curva de calibración sensor de humedad*



Con los datos obtenidos por ambos sensores se realiza el diagrama de dispersión para comprobar si las mediciones realizadas por ambos dispositivos se encuentran relacionadas. Mediante la Figura 58, se establece que las mediciones presentan una tendencia idéntica, el coeficiente de correlación lineal tiene un valor de 0.962 muy cercano a 1 lo que exhibe un tipo de correlación fuertemente positiva.

**Figura 58**

*Diagrama de dispersión mediciones de temperatura*



### **Configuración del sensor de nivel**

El sensor ultrasónico tampoco requiere de un proceso previo de calibración debido a que su configuración es realizada en base a programación. Para realizar las mediciones se conectan las salidas trigger y echo en los pines análogos del microcontrolador. El sensor ultrasónico realiza la medición del tiempo que se tarda en enviar y recibir un pulso sonoro. Se conoce que el sonido tiene una velocidad de  $343 \text{ m/s}$ , para realizar las mediciones en  $\text{cm/us}$  se realiza la conversión de unidades y se obtiene que el sonido recorre un  $\text{cm}$  en  $29.2 \text{ us}$ . La ecuación (15) permite establecer el nivel del contenedor en base al tiempo que requiere la emisión y recepción de un pulso.

$$\text{Nivel (cm)} = \frac{\text{Tiempo (us)}}{2 \times 29.2 \text{ us}} \quad (15)$$

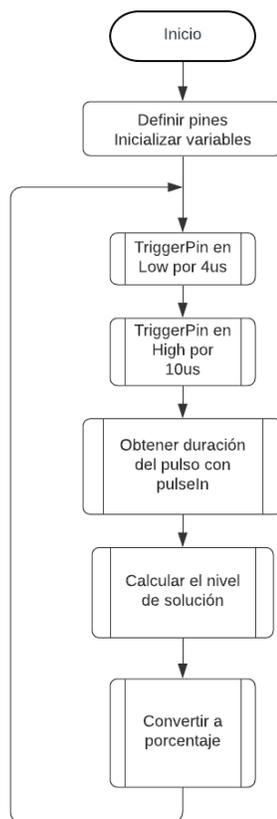
La razón por la que se realiza la división entre dos, es porque se requiere medir el tiempo que se demora en ir y retornar el pulso emitido.

### Desarrollo del software

Para realizar el proceso de medición de nivel de sustancia nutritiva, en el respectivo sketch de configuración se definen los pines de entrada analógica a utilizar, luego se requiere generar un pulso en el disparador "Trigger" de por lo menos 10  $\mu s$ , pero previamente se coloca a este pin en bajo por 4  $\mu s$  para garantizar un disparo nítido. Por medio de la función pulseIn se obtiene el tiempo que requiere el pulso para retornar al sensor y luego se calcula la distancia en base a la ecuación (15).

### Figura 59

*Diagrama de flujo para el sensor de nivel*

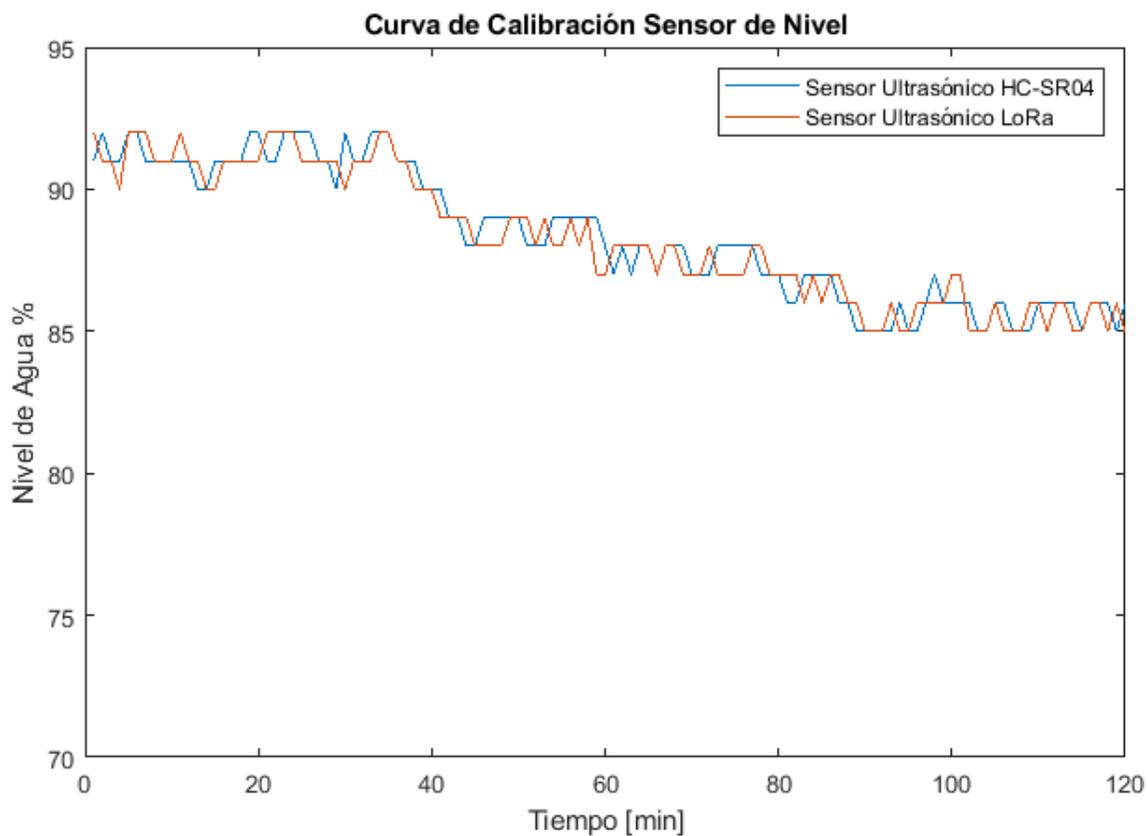


### Respuesta del sensor ultrasónico

Se realiza mediciones tanto con el equipo patrón como con el sensor. Las pruebas se efectúan en el contenedor de solución nutritiva por un tiempo de 120 minutos, los resultados de las curvas de medición se observan en la Figura 60, el comportamiento del sensor ultrasónico a lo largo del tiempo se ajusta a la curva obtenida por el otro sensor utilizado para la verificación de la calidad de los datos medidos.

**Figura 60**

*Curva de calibración sensor de humedad*

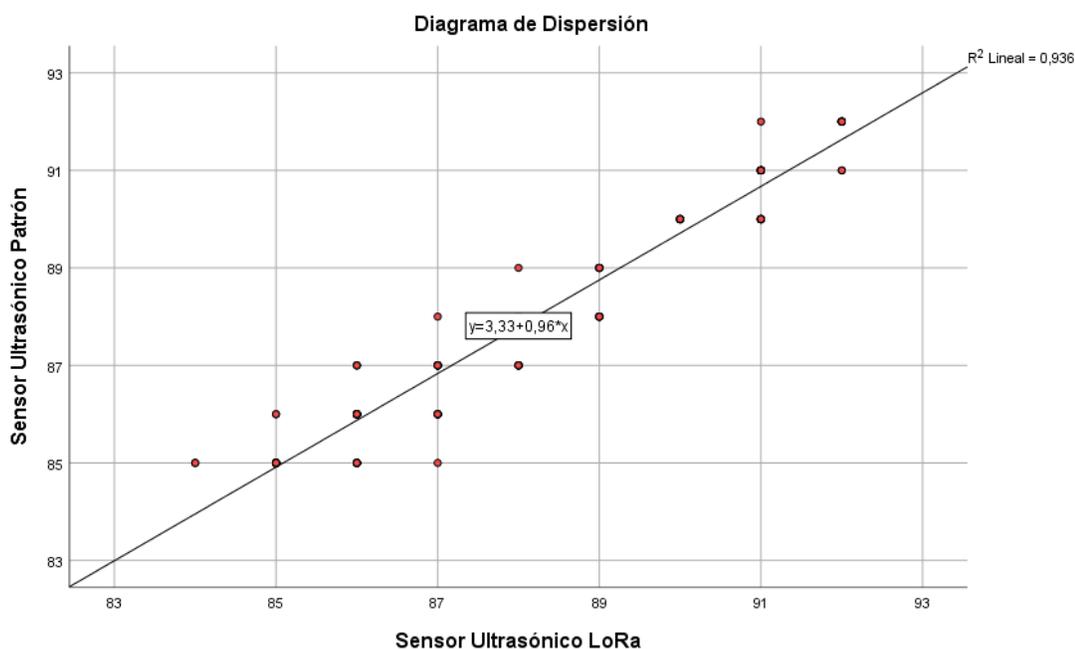


Con los datos obtenidos por ambos sensores se realiza el diagrama de dispersión para comprobar si las mediciones realizadas por ambos dispositivos se

encuentran relacionadas. Mediante la Figura 61, se establece que las mediciones presentan una tendencia idéntica, el coeficiente de correlación lineal tiene un valor de 0.936 muy cercano a 1 por lo que se concluye que existe una correlación fuertemente positiva.

**Figura 61**

*Diagrama de dispersión mediciones de temperatura*



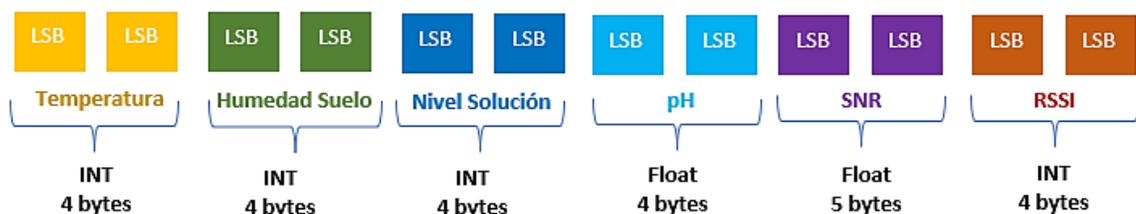
### ***Programación del Nodo Final***

LoRa *Shield* constituye el núcleo del Nodo Final, para el desarrollo del software se realiza previamente la calibración y programación de los sensores de la red, se recepta las lecturas y se realiza un procesamiento de los datos para el envío por medio de LoRa hacia el *Gateway*. Para la configuración se toma en consideración los siguientes aspectos:

1. Realizar la importación de librerías, inicializar la frecuencia, definir el nodo Id y crear funciones para transmitir los datos.
2. Efectuar una concatenación de todas las variables con el router id que permita la identificación del nodo transmisor que envía la información.
3. La carga útil está constituida de los datos de mediciones provistas por los sensores de la red de monitorización y su formato se observa en la Figura 62.
4. Basado en la experimentación, se concluye que el agua tarda aproximadamente 10 minutos en retornar al contenedor de la solución, la bomba es automatizada para que permanezca encendida durante 5 minutos y realice la distribución de solución nutritiva por todos los tubos del sistema. Por este motivo el Nodo Final realiza las transmisiones de datos cada 15 minutos para verificar que los parámetros de pH y nivel de solución sean los óptimos antes de realizar la recirculación de agua y asimismo que la humedad del suelo y temperatura sea la propicia para la germinación de las semillas.

**Figura 62**

*Carga útil formado por las variables de monitorización*

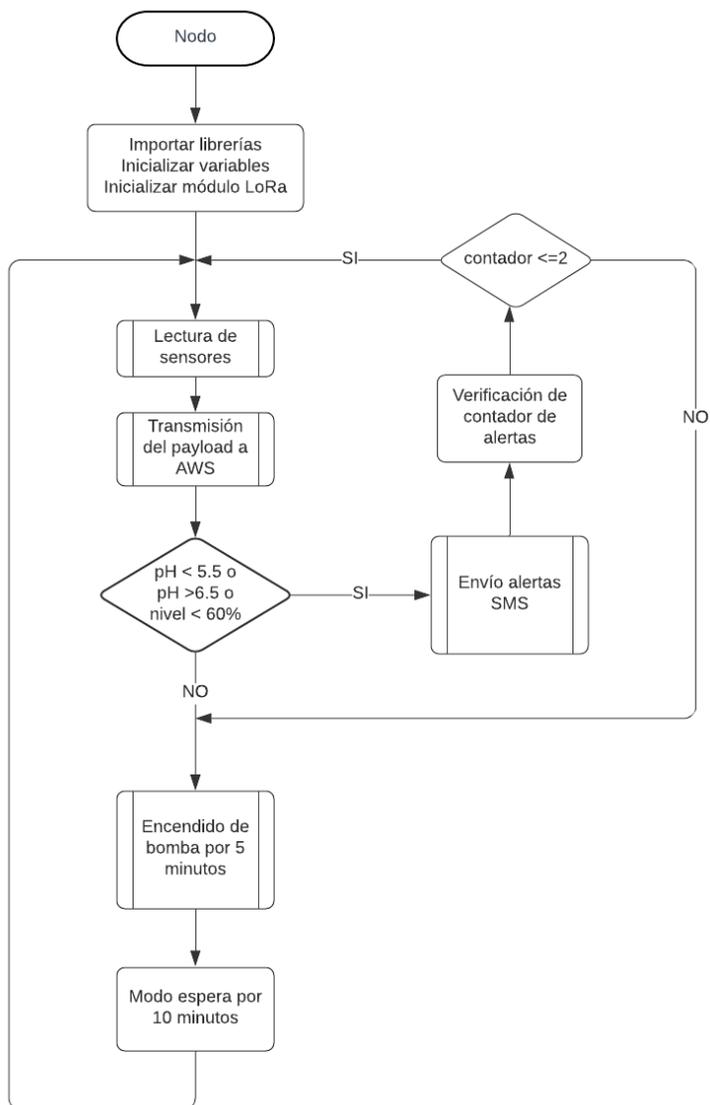


*Nota.* La figura expone el tipo de dato, los nombres de asignación y el número de bytes requeridos para almacenar los datos de las variables físicas a transmitir.

Por medio del diagrama de flujo de la Figura 63, se realiza la programación de la placa LoRa Shield con el lenguaje C++ en el software Atom.

Figura 63

Diagrama de flujo del sistema de monitorización



Para iniciar el proceso de comunicación se importa la librería LoRa.h, se declara la variable que será enviada hacia el *Gateway* y que almacena tanto el nodo Id, así como los datos provistos por los sensores. Esta variable es definida como un vector entero de 8 bits, luego se realiza la configuración de comunicación con LoRa en la cual se establece el ancho de banda, el factor de dispersión, la tasa de código y las palabras

de sincronización para una red pública como es 0x34 para dispositivos SX127x, como se observa a continuación:

### Figura 64

#### Configuración del Nodo Final

```

#include <LoRa.h>
char *node_id = "<4567>";
uint8_t datasend[54];

Serial.begin(9600);
while (!Serial);
Serial.println(F("Iniciando Comunicacion"));
if (!LoRa.begin(915000000))
{
    Serial.println("Comunicación con LoRa fallida!");
    while (1);
}

LoRa.setSpreadingFactor(7);
LoRa.setSignalBandwidth(125000);
LoRa.setCodingRate4(5);
LoRa.setSyncWord(0x34);

Serial.println("Comunicación con LoRa exitosa!!...");

```

Posteriormente, se crearon las funciones de la Tabla 31 para obtener los datos de sensado y en la función *mediciones ()* se realiza la conversión de cada variable a ASCII por medio de *dtostrf ()* para almacenarla como una cadena. *dtostrf ()* es una función que requiere de cuatro parámetros: el primero es la variable a convertir, el segundo una variable *char* que establece el número de dígitos, el tercero una variable *char* que determina el número de dígitos luego del decimal y el último parámetro es la variable que almacena la conversión.

Se utiliza la función *strcat ()* para concatenar en la variable *data* el valor del nodo Id con los datos de cada uno de los sensores, más algunos elementos como comas y

espacios que hacen más fácil el proceso de decodificación. Finalmente, por medio la función `strcpy ()` se arma el payload al copiar la cadena concatenada en la variable `datasend`.

```

1  dtostrf (tem,0,1,tem_1);
2  dtostrf (hum,0,1,hum_1);
3  dtostrf (total,0,1,nivel);
4  dtostrf (totalph,0,2,ph);
5  dtostrf (SNR1,0,2,snr_1);
6  dtostrf (RSSI1,0,0,rssi_1);
7
8  strcat (data,"");
9  strcat (data,tem_1);
10 strcat (data,",");
11 strcat (data,"");
12 strcat (data,nivel);
13 strcat (data,",");
14 strcat (data,"");
15 strcat (data,ph);
16 strcat (data,",");
17 strcat (data," ");
18 strcat (data,hum_1);
19 strcat (data,",");
20 strcat (data,snr_1);
21 strcat (data,",");
22 strcat (data,rssi_1);
23 strcat (data,"");
24 strcpy ((char *)datasend,data);

```

Por último, para enviar la carga útil al *Gateway* se utiliza las siguientes funciones:

```

1 void SendData ()
2 { mediciones ();
3   LoRa.beginPacket ();
4   LoRa.print ((char *)datasend);
5   LoRa.endPacket ();
6 }

```

Para la decodificación de mensajes recibidos por el *Gateway* se hace uso de la función `onReceive()` en la cual se realiza el proceso de encendido y apagado automático de la bomba del sistema. Cuando se recibe un mensaje se realiza la conversión a char, `on2` y `off2` son los mensajes transmitidos desde la plataforma IoT,

mientras que *on* y *off* son mensajes transmitidos por el servidor con el uso de Node.JS, posteriormente se realiza comparaciones en base al siguiente código:

```

1 void onReceive(int packetSize) {
2   if (packetSize == 0) return;
3   String mensajeRX = "";
4
5   while (LoRa.available()) {
6     mensajeRX += (char)LoRa.read();
7   }
8
9   if(mensajeRX == "off2"){
10    // APAGAR INDEFINIDAMENTE
11    digitalWrite(8, LOW);
12    Serial.println("Bomba apagada");
13  }
14
15  if(mensajeRX == "on"){
16    // ENCENDER BOMBA
17    Serial.println("Prendiendo Bomba...");
18    digitalWrite(8, HIGH);
19    delay(300000); //5 minutos encendido
20    Serial.println("Apagando Bomba...");
21    digitalWrite(8, LOW);
22  }
23 }

```

Para realizar las mediciones de los parámetros de *SNR* y *RSSI*, se hace uso de las librerías de LoRa como se presenta a continuación:

```

1 float relacionSNR() {
2   float snr1 = LoRa.packetSnr();
3   return snr1;
4 }
5
6 int relacionRSSI() {
7   int RSSI1 =LoRa.packetRssi();
8   return RSSI1;
9 }

```

### ***Carcasa del Nodo Final***

Previo al diseño de la carcasa se toma en consideración las recomendaciones planteadas en (OMM, 2018), donde se establece parámetros y propiedades que deben

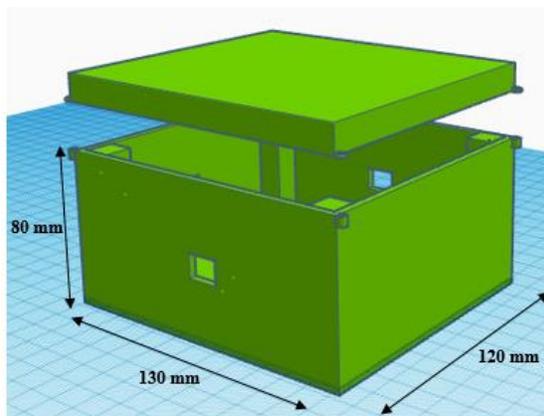
cumplir las carcasas de dispositivos orientados a la adquisición de datos. Las principales sugerencias son:

- El material debe ser resistente, de preferencia plástico o madera.
- El color de la carcasa debe ser blanco o pintada con el propósito de reflejar la radiación.
- Disponer de ventilaciones y cubierta.
- El dispositivo debe ser colocado en el rango de 1.5 a 2 metros de altura para la toma de datos.

Basado en las recomendaciones de la OMM se procede a diseñar la carcasa en Tinkercad, software en línea gratuito que permite realizar procesos de modelado en tres dimensiones. El diseño se efectúa según las dimensiones de la placa LoRa Shield, el módulo convertidor y la Shield de borneras. El material de la carcasa es de plástico resistente a condiciones climáticas adversas como fuertes lluvias o elevadas temperaturas, en la Figura 65 se observa el diseño final de la carcasa.

### Figura 65

*Carcasa del Nodo Final*



*Nota.* La figura expone las dimensiones de la carcasa diseñada para el Nodo Final.

### **Frecuencia de Operación**

Para el desarrollo de la red de monitorización se utiliza el plan de frecuencias AU 915-928 MHz cuya canalización es especificada en la Tabla 8. En base a las consideraciones expuestas por la *ARCOTEL* en la resolución aprobada en diciembre del 2017, se manifiesta: "En las bandas 915 – 928 MHz, 2 400 – 2 483,5 MHz, 5 150 – 5 350 MHz, 5 470 – 5 725 MHz y 5 725 – 5 850 MHz y 24,05 – 24,25 GHz operan, a título secundario, sistemas que ocupan espectro radioeléctrico para uso determinado en Bandas Libres (UDBL), para los servicios fijo y móvil". (ARCOTEL, 2017)

### **Configuración RF de LoRa Shield**

Para poder comunicar LoRa Shield con el *Gateway* LG01-N de un solo canal, se requiere hacer modificaciones en el archivo *config* de la librería LMIC para configurar los parámetros de la capa de RF analizados en el capítulo 3. El archivo a modificar se encuentra ubicado en la ruta: *Documents\Arduino\libraries\arduino-lmic-master\src\lmic*. La configuración realizada se observa en la Figura 66.

### **Figura 66**

#### *Configuración de RF*

```

// #define CFG_eu868 1
#define CFG_us915 1 // Frecuencia de Operación en Ecuador
// #define CFG_au921 1
// #define CFG_as923 1
// #define CFG_in866 1

#define LG02_LG01 1 // Comunicación con el tipo de Gateway

// US915: DR_SF10=0, DR_SF9=1, DR_SF8=2, DR_SF7=3, DR_SF8C=4
// DR_SF12CR=8, DR_SF11CR=9, DR_SF10CR=10, DR_SF9CR=11, DR_SF8CR=12
#if defined(CFG_us915) && defined(LG02_LG01)
// CFG_us915 || CFG_as923
// #define LG02_UPFREQ 902320000
// #define LG02_DNWFREQ 902320000
#define LG02_UPFREQ 915000000 // Frecuencia de Uplink
#define LG02_DNWFREQ 923300000 // Frecuencia de Downlink
#define LG02_RXSF 3 // Factor de Dispersión de 7: DR_SF7

```

### Configuración del Gateway LG01-N

Para iniciar el proceso de configuración se ingresa a la interfaz de Dragino, por medio de un cable de red se conecta el *Gateway* a un puerto LAN de la PC. Luego desde cualquier navegador se ingresa a la IP 10.130.1.1 y se escribe las credenciales de la Figura 69 para iniciar una sesión Web:

**Figura 67**

*Credenciales de sesión Dragino*

Por favor, inicia sesión

http://10.130.1.1

Tu conexión con este sitio web no es privada

Usuario: root

Contraseña: ..... dragino

Iniciar sesión Cancelar

Luego en la sección LoRa se realiza la configuración de RF como se observa en la Figura 68, en la cual se elige el factor de propagación, la frecuencia, la longitud del preámbulo, la tasa de codificación, el ancho de banda y la potencia.

**Figura 68**

*Configuración de parámetros LoRa en el Gateway*

### LoRa Configuration

Debug Level

**Radio Settings**

Frequency (Hz)	<input type="text" value="915000000"/>	RF Bandwidth (Hz)	<input type="text" value="125kHz"/>
Spreading Factor	<input type="text" value="SF7"/>	Coding Rate	<input type="text" value="4/5"/>
Preamble Length	<input type="text" value="8"/>	LoRa Sync Word	<input type="text" value="52"/>
RF Power (0-20) dBm	<input type="text" value="20"/>		

Se configura los parámetros de *MQTT* previamente analizados en el capítulo 3 como se observa en la Figura 69. Se define la dirección DNS del bróker, el User ID, Client ID, el puerto del bróker. Se habilita Publish para la transmisión de datos hacia los dispositivos suscriptores, se selecciona la Calidad de Servicio, el formato del tópic y el formato de los datos. Se habilita Subscribe para la recepción de mensajes desde la plataforma bajo el tópic *sensorp*.

**Figura 69**

*Configuración MQTT en el Gateway*

### MQTT Client Configuration

MQTT Server Profile	<input type="text" value="General"/>		
Broker Address [-h]	<input type="text" value="wirelbits.ga"/>	Broker Port [-p]	<input type="text" value="1883"/>
User ID [-u]	<input type="text" value="User ID"/>	Password [-P]	<input type="text" value="Password"/> <a href="#">Show</a>
Certificate [--cert]	<input type="text"/>	Key [--key]	<input type="text"/>
CA File [--cafile]	<input type="text"/>		
Client ID [-i]	<input type="text" value="dragino_gateway"/>		

**Publish**

Enable Publish <input checked="" type="checkbox"/>	Quality of Service [-q] <input type="text" value="QoS 0"/>	Topic Format [-t] <input type="text" value="sensores"/> ?
		Data Format [-m] <input type="text" value="DATA"/> ?

**Subscribe**

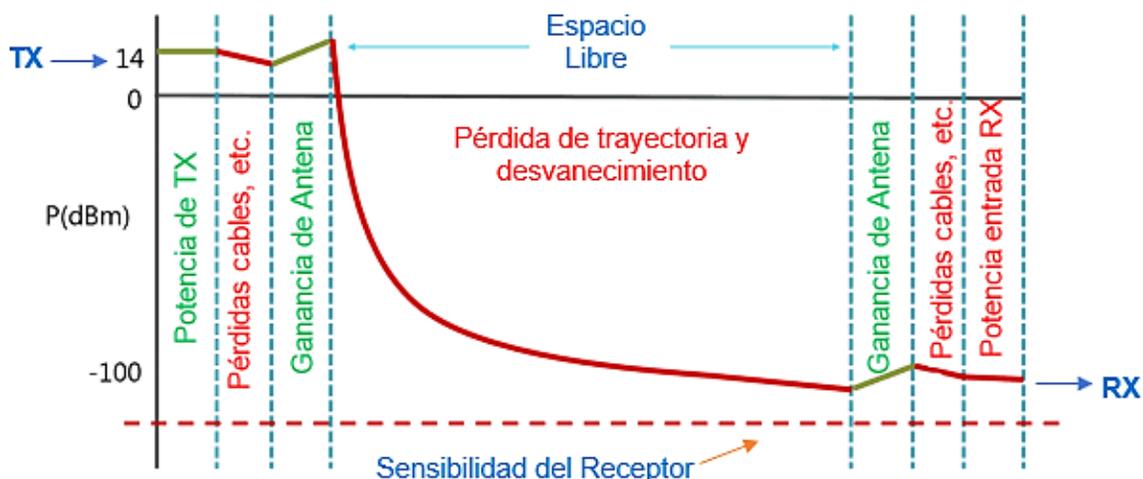
Enable Subscribe <input checked="" type="checkbox"/>	Quality of Service [-q] <input type="text" value="QoS 0"/>	Topic Format [-t] <input type="text" value="sensorp"/> ?
--	--	--

### ***Cálculo del área de cobertura***

Con la finalidad de disponer de una noción del área de cobertura que se espera alcanzar con los parámetros de RF configurados en el *Gateway* LoRa, se realiza el cálculo del radio máximo que está en función del presupuesto del enlace.

Figura 70

Presupuesto del enlace



En redes inalámbricas, el presupuesto del enlace (*Link Budget*) representado en la Figura 70, es la cuantificación de las ganancias y pérdidas producidas en el transmisor, canal de propagación y receptor. Para calcular el *Link Budget* se necesita calcular la pérdida máxima por el trayecto denominada *Path Loss* definida por la ecuación (16):

$$Path Loss (dB) = -P_{RX} + P_{TX} + G_{AT} + G_{AR} - L_{con} \quad (16)$$

donde:  $P_{RX}$  constituye la potencia de recepción,  $P_{TX}$  la potencia de transmisión,  $G_{AT}$ ,  $G_{AR}$  las ganancias de las antenas de transmisión y recepción y  $L_{con}$  las pérdidas provocadas por los conectores. La Potencia de Recepción establecida en la ecuación (17), depende de la sensibilidad en el receptor y el Margen de desvanecimiento ( $MD$ ):

$$P_{RX}(dBm) = Sensibilidad_{RX} (dBm) + MD \quad (17)$$

La sensibilidad de LoRa, se puede calcular por medio de la ecuación (18):

$$Sensibilidad_{RX} (dBm) = -174 + 10\log(BW) + SNR + NF \quad (18)$$

Una vez configurado los parámetros de RF, el ancho de banda ( $BW$ ) utilizado es de 125 kHz, la relación señal ruido ( $SNR$ ) en base a la Tabla 6 para un  $SF$  de 7 es de -7.5 dB y la figura de ruido ( $NF$ ) típica para *LoRa* es de 6 dB, por medio de la ecuación (18) se establece:

$$Sensibilidad_{RX} = -174 + 10\log(125) - 7.5 + 6 = -154.53 \text{ (dBm)}$$

Se utilizan los parámetros configurados en los equipos como: la banda de frecuencia de 915 MHz, según el manual de los equipos establecidos en la *FCC ID*, la potencia de transmisión es de +20 dBm, las antenas de transmisión y recepción tienen una ganancia de 2 dBi cada una, se considera una pérdida por conectores y cables en todo el sistema de 1.5 dB y un margen de desvanecimiento de 30 dB (disponibilidad de 99.9%), se define el *Path Loss* y  $P_{RX}$  (dBm):

$$P_{RX} (dBm) = -154.53 + 30 = -124.53$$

$$Path Loss (dB) = -(-124.53) + 20 + 2 + 2 - 1.5 = 147.03 \text{ dB}$$

En la realidad, en ciudades urbanas y semiurbanas no existe Espacio Libre (*Free Space*) debido a que se tiene diversas pérdidas ocasionadas por obstáculos como edificios y árboles que dan origen a fenómenos como: multitrayecto, desvanecimientos, efecto Doppler, etc. Por este motivo para establecer el *Path Loss* se considera una pérdida adicional de 15 dB generada por estructuras de concreto (ITU, 2016). Por consiguiente, se tiene que:

$$Path Loss (dB) = 147.03 - 15 = 132.03 \text{ (dB)}$$

Para establecer el radio de cobertura, se utiliza el valor de *Path Loss* calculado y se aplica el modelo de propagación Okumura-Hata Cost 231, que opera en un rango de

frecuencias de 800 a 2000 MHz. Este modelo viene definido matemáticamente por la ecuación (19).

$$L(dB) = 46.3 + 33.9 \times \log_{10}(f) - 13.82 \times \log_{10}(h_b) - a(h_m) + K_a \quad (19)$$

Donde se tiene que:

$$K_a = [44.9 - 6.55 \times \log_{10}(h_b)] \times \log_{10}(d) \quad (20)$$

- Frecuencia de operación ( $f$ ): 915 MHz
- Altura de la antena del *Gateway* con respecto al suelo ( $h_b$ ): 10.5 m
- Altura del nodo final con respecto al suelo ( $h_m$ ): 1.5 m
- Distancia del *Gateway* al nodo final ( $d$ )
- Factor de corrección  $a(h_m)$ : para ciudades medianas como Sangolquí este valor de se define mediante la ecuación (21).

$$a(h_m) = [1.1 \times \log_{10}(f) - 0.7] \times h_m - 1.56 \times \log_{10}(f) + 0.8 \quad (21)$$

Al aplicar las ecuaciones anteriormente establecidas se calcula lo siguiente:

$$a(h_m) = [1.1 \times \log_{10}(915) - 0.7] \times 1.5 - 1.56 \times \log_{10}(915) + 0.8 = \mathbf{0.02}$$

$$K_a = [44.9 - 6.55 \times \log_{10}(10.5)] \times \log_{10}(d) = \mathbf{38.21 \times \log_{10}(d)}$$

$$L(dB) = 46.3 + 33.9 \times \log_{10}(915) - 13.82 \times \log_{10}(10.5) - 0.02 + K_a$$

$$L(dB) = 132.56 + 38.21 \times \log_{10}(d)$$

Al reemplazar el valor del Path Loss, se define la distancia, equivalente al radio máximo teórico de cobertura con un SF igual a 7:

$$132.03 = 132.56 + 38.21 \times \log_{10}(d)$$

$$\mathbf{d = 0.96 km}$$

$$\mathbf{A = 2.89 km^2}$$

### Simulación del área cobertura

Luego de realizar el cálculo teórico del área de cobertura, se empleó el software de simulación Radio Mobile para generar una estimación de cobertura en base al tipo de terreno, condiciones del clima, vegetación, características del equipo de propagación. La simulación se realiza en función de las coordenadas de ubicación del *Gateway* y parámetros de la Tabla 32.

**Tabla 32**

*Parámetros de configuración en Radio Mobile*

Parámetro	Valor
<b>Latitud</b>	00° 20' 30.82"S
<b>Longitud</b>	78° 26' 58.15"W
<b>Altura de la antena Tx</b>	10.5 m sobre el suelo
<b>Tipo de antena</b>	Omnidireccional
<b>Frecuencia</b>	915 MHz
<b>Potencia Tx</b>	100 mW
<b>Altura de la antena Rx</b>	1.5 m sobre el suelo

*Nota.* El método de cálculo utilizado por Radio Mobile es el Rec. UIT-R P.256-11

Los resultados de simulación en la Figura 71, permiten identificar que zonas cercanas al *Gateway* presentan un color verde y tienen potencias de recepción de -103 dBm lo que garantiza una óptima comunicación entre diversos nodos finales y el *Gateway*. Las zonas alejadas se representan de color amarillo en las cuáles se tiene valores de potencia de señal superiores a -113 dBm, en estas zonas no se garantiza una cobertura óptima de señal. El software también permite definir la densidad de



múltiples servicios computacionales virtuales los cuáles se cobran únicamente por lo que se utiliza. Amazon EC2 provee de una gran variedad de tipos de instancia que se adaptan a las necesidades de ejecución de aplicaciones, permite elegir la capacidad del CPU, memoria, red y capacidad de almacenamiento.

Previo a la configuración de la instancia, se elige una *AMI* (del inglés Amazon Machine Image) que constituye una plantilla con configuraciones de software (sistema operativo y servidor de aplicaciones), componentes necesarios para la ejecución de instancias. El sistema operativo escogido es la versión de Ubuntu 18.04 LTS de 64 bits. En la Figura 74, se observa los diferentes tipos de instancias, el servidor virtual elegido es un t2 micro, con 1 vCPU de 2.5 GHz, una memoria de 1 GiB, almacenamiento *EBS* (del inglés Elastic Block Store) y con un desempeño de red de bajo a moderado.

## Figura 72

### Selección de la instancia

Familia	Tipo	vCPU ⓘ	Memoria ( GiB)
t2	t2.nano	1	0.5
t2	t2.micro Apto para la capa gratuita	1	1
t2	t2.small	1	2
t2	t2.medium	2	4
t2	t2.large	2	8
t2	t2.xlarge	4	16
t2	t2.2xlarge	8	32

*Nota.* La instancia seleccionada tiene compatibilidad con IPv6.

Luego se procede a configurar el dispositivo de almacenamiento, el tamaño elegido es de 25 GiB de tipo *SSD* de uso general con un *IOPS* (del inglés Input/Output Operations per Second) de 100/3000 por segundo.

## Figura 73

### Configuración del almacenamiento

Dispositivo ⓘ	Snapshot ⓘ	Tamaño (GiB) ⓘ	Tipo de volumen ⓘ	IOPS ⓘ
/dev/sda1	snap-0b8534b77af28fde5	25	SSD de uso general (gp2) ▾	100/3000

AWS por defecto cierra todos los puertos por temas de seguridad informática. Por ello se requiere configurar los grupos de seguridad (security groups) que actúan como firewall virtual y disponen de reglas para los tipos de protocolos que se permiten, el rango de puertos y el control del tráfico de entrada y salida. Cuando no se define un grupo de seguridad, Amazon EC2 ejecuta el security group predeterminado. En el proceso de configuración y lanzamiento de la instancia se puede crear uno o varios security groups, el grupo de seguridad configurado en la instancia se observa en la Tabla 33.

**Tabla 33**

### Security group

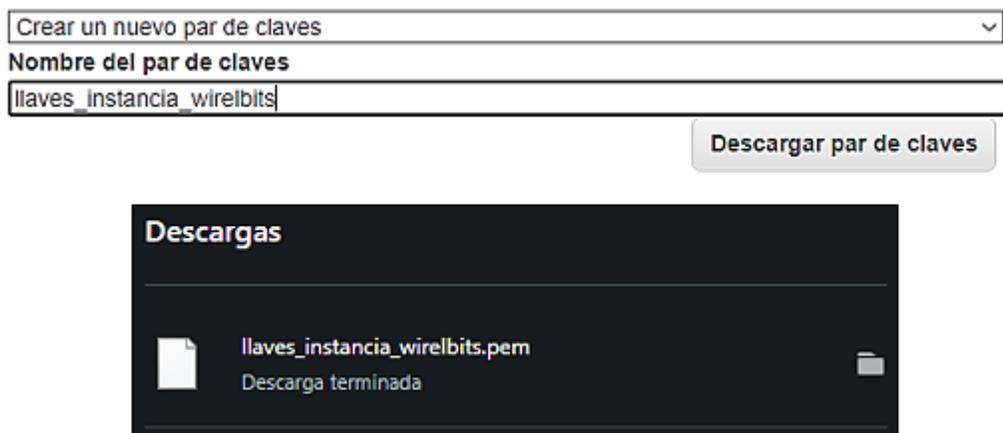
Tipo	Protocolo	Rango de Puertos	Descripción
HTTP	TCP	80	HTTP
Regla TCP	TCP	8083	Panel CP
Regla TCP	TCP	18093	EMQXs
Regla TCP	TCP	18083	EMQX
Regla TCP	TCP	8883	EMQ
Regla TCP	TCP	8090	EMQ
Regla TCP	TCP	1883	EMQ
HTTPS	TCP	443	HTTPS
ICMP IPv4	TCP	N/D	PING

<b>Regla TCP</b>	Regla TCP	8093	EMQ
<b>Regla TCP</b>	TCP	12000-12100	FTP
<b>SSH</b>	TCP	22	SSH
<b>Regla TCP</b>	TCP	21	FTP
<b>Regla TCP</b>	TCP	8094	EMQ
<b>MYSQL/Aurora</b>	TCP	3306	SQL

Finalmente, se crea un par de claves que permiten la conexión segura vía SSH hacia la instancia. Se debe descargar el archivo de las claves privadas y guardarlo de manera segura, ya que AWS permite la descarga del archivo sólo una vez. En la figura 74 se observa la creación de las llaves de la instancia y la descarga del archivo.

#### Figura 74

*Llave de la instancia*

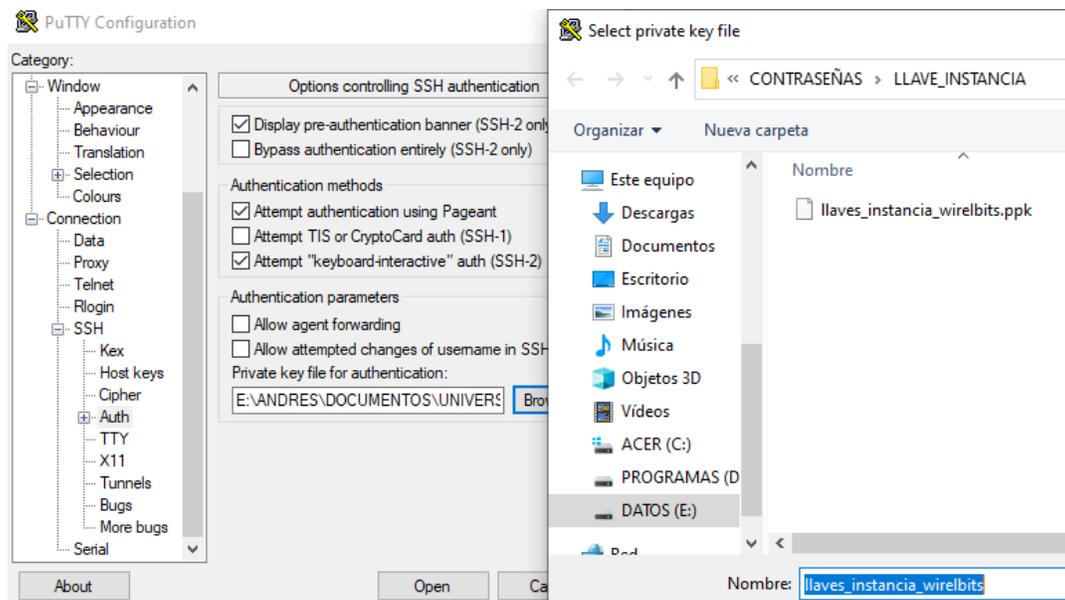


#### Conexión SSH

Una vez descargado el archivo `.pem`, para conectarse a la instancia por medio de SSH con el software Putty se necesita realizar la conversión al formato `.PPK` (del inglés Putty Private Key). Para ello se utiliza la herramienta PUTTYgen que forma parte del paquete Putty.

## Figura 75

### Conversión del archivo .pem



Una vez realizada la conversión, en la pestaña *Auth* se importa la llave privada *.PPK* y se procede a realizar la conexión a la instancia de manera remota. Para este proceso se utiliza la dirección IP o Hostname y el puerto configurado en la Tabla 33 que posibilita la escucha de la conexión SSH. En la Figura 76, se observa la conexión a la instancia con la utilización del software Putty.

## Figura 76

Conexión vía SSH a la instancia

```
ubuntu@ip-172-31-42-90: ~  
  
System information as of Mon Jun  7 18:16:57 UTC 2021  
  
System load:  0.0          Processes:      92  
Usage of /:   4.7% of 24.17GB  Users logged in:  0  
Memory usage: 19%          IP address for eth0: 172.31.42.90  
Swap usage:   0%  
  
0 packages can be updated.  
0 of these updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@ip-172-31-42-90:~$
```

### Configuración del dominio y certificados SSL

El nombre de dominio se define como la cadena de texto que corresponda a una dirección IP específica, es decir el texto que cualquier usuario escribe en un navegador para acceder a un sitio web. Se utiliza el proveedor *Freenom*, que facilita dominios gratuitos de última tecnología para el desarrollo de sitios web. Freenom utiliza la tecnología de la nube *Anycast*, garantiza estabilidad y un rendimiento óptimo de sus dominios que provee. En la Tabla 34, se observa la configuración del dominio realizada en Freenom.

**Tabla 34***Configuración de dominio*

Type	TTL	Target
<b>A</b>	3600	15.228.74.184
<b>CNAME</b>	3600	wirelbits.ga

*Nota.* Los registros A, también conocidos como registros de host permiten asociar una dirección IP a un nombre de dominio, el registro canónico o CNAME permite realizar el enlace de un nombre de alias con un nombre de dominio auténtico y *TTL* (del inglés Time To Live), se define como el tiempo en segundos que transcurren antes de aplicarse los cambios realizados en los registros.

Una vez realizada la correcta asignación de la IP de la instancia con el nombre de dominio, en la Figura 77 se observa el acceso al sitio web desde un navegador. El sitio no cuenta con la tecnología estándar SSL por lo cual no existe un cifrado de tráfico de los datos.

**Figura 77***Plataforma IoT sin certificados SSL*

El certificado SSL es requerido por la plataforma web IoT para preservar la seguridad de los datos transmitidos por la red de sensores, verificar la propiedad del sitio web que evita que hackers creen una versión falsa del mismo y transmitir confianza entre los usuarios que utilizarán la plataforma. En la Figura 78, se observa la instalación de los certificados SSL.

### Figura 78

#### Instalación de certificados SSL

The screenshot shows a configuration interface for SSL. It includes the following sections:

- SSL Support**: A checked checkbox.
- Lets Encrypt Support**: A checked checkbox.
- SSL Home**: A dropdown menu with the value "public\_html".
- SSL Certificate**: A text area containing the following text:
 

```
-----BEGIN CERTIFICATE-----
MIIGMDCCBRigAwIBAgISBLNV8QD77AN/zghwHRO49qchMA0
GCSqGSIb3DQEBCwUA
MDIx CzAJBgNVBAYTAIVTMRYwFAYDVQQKEw1MZXQncyBFb
-----
```
- SSL Key**: A text area containing the following text:
 

```
-----BEGIN RSA PRIVATE KEY-----
MIJKAIBAAKCAgEAuvQUYA4q3ZCdUCn23TxwscKITaGsv1ruT
W/8eaJuwRDGiMLr
8xzAmFEJ2EgILGG7hTPY6Y6C+K1yd004btKMbLewPtNArQFst
-----
```
- SSL Certificate Authority / Intermediate (optional)**: A text area containing the following text:
 

```
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFjCCAv6gAwIBAgIRAJErCErPDBinU/bWLiWnX1owDQYJK
-----
```

*Nota.* La figura presenta la configuración de los certificados SSL y la carpeta de ubicación de los mismos.

Una vez realizada la configuración de los certificados, se conecta a la instancia de manera remota y se accede a la carpeta `conf/web` para verificar la existencia de los archivos (Figura 79).

## Figura 79

### Certificados SSL

```

root@wirelbits:/home/admin/conf/web# ls
nginx.wirelbits.ga.conf_letsencrypt  ssl.wirelbits.ga.pem
snginx.wirelbits.ga.conf_letsencrypt  wirelbits.ga.apache2.conf
ssl.wirelbits.ga.ca                  wirelbits.ga.apache2.ssl.conf
ssl.wirelbits.ga.crt                  wirelbits.ga.nginx.conf
ssl.wirelbits.ga.key                  wirelbits.ga.nginx.ssl.conf

```

En la Figura 80 se observa que aparece un ícono de candado en el buscador del navegador, este permite realizar la consulta de la información del certificado SSL instalado, también es garantía que el sitio web cuenta con *HTTPS* (del inglés *Hyper Text Protocol Secure*) que garantiza la conexión de manera segura entre el servidor y los usuarios.

## Figura 80

### Plataforma IoT con certificados SSL



## Configuración del bróker MQTT

Luego de realizar varias pruebas con diferentes brókeres, se opta por EMQ que es un bróker de mensajería IoT MQTT de código abierto, excelente para desarrollar aplicaciones en tiempo real, con una latencia baja y basado en el patrón publish-subscribe. EMQ es estable y permite múltiples conexiones de clientes a gran escala,

millones de conexiones son soportadas por un solo nodo servidor. Dispone de diferentes extensiones y pluggins para autenticación y otras funcionalidades. Previo al proceso de instalación es importante instalar algunas de las siguientes dependencias requeridas:

```
1 sudo apt update && sudo apt install -y \
2 apt-transport-https \
3 ca-certificates \
4 curl \
5 gnupg-agent \
6 software-properties-common
```

La versión de EMQ a utilizar es la V4.3.5, luego mediante la herramienta *wget* se descarga el paquete de software para Ubuntu, se instala y se ejecuta el bróker desde la terminal de consola de Putty con los siguientes comandos:

```
1 wget https://www.emqx.com/en/downloads/broker/4.3.8/emqx-ubuntu18.04-
2 4.3.8-amd64.deb
3 sudo apt install ./emqx-ubuntu18.04-4.3.8-amd64.deb
sudo emqx start
```

Después de la instalación, se detiene el bróker con el comando *emqx stop* y se cambia los puertos de escucha para un correcto funcionamiento. Los puertos a cambiar son el *listener.ws.external* del puerto 8083 al 8093 y *management.listener.http* del puerto 8080 a 8090. También se configura el número máximo de clientes y el tiempo de espera para transmitir una respuesta, la configuración realizada se observa en la Figura 81.

## Figura 81

### Configuración de EMQ

```

## Examples: 8083, 127.0.0.1:8083, :::8083
listener.ws.external = 8093

## The path of WebSocket MQTT endpoint
##
## Value: URL Path
listener.ws.external.mqtt_path = /mqtt

## HTTP Listener

management.listener.http = 8090
management.listener.http.acceptors = 2
management.listener.http.max_clients = 512
management.listener.http.backlog = 512
management.listener.http.send_timeout = 15s
management.listener.http.send_timeout_close = on

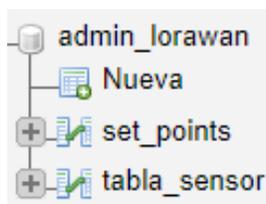
```

## Base de datos

La base de datos permite el almacenamiento y acceso a la información de las variables físicas que se encuentran relacionados entre sí, el ID de una tabla conocido como clave facilita la identificación de cada fila de manera que la plataforma web acceda a esta información y pueda ser mostrada a los usuarios. El nombre de la base creada es *admin\_lorawan*, se encuentra alojada en la instancia del servidor virtual creado en AWS y está compuesta de dos tablas como se observa a continuación:

## Figura 82

### Base de datos



1. *tabla\_sensor*: en esta tabla se guarda todas las variables físicas monitorizadas, la fecha y hora en la que se realiza la inserción
2. *set\_points*: permite el almacenamiento de los valores seteados por los usuarios, los mismos que sirven para comparar con los datos transmitidos por los sensores y emitir alertas SMS cuando sea necesario.

La estructura de las tablas se observa en la Figura 83, en la cual se identifica los nombres de asignación de variables, el tipo y los atributos de los datos.

**Figura 83**

*Tabla de datos*

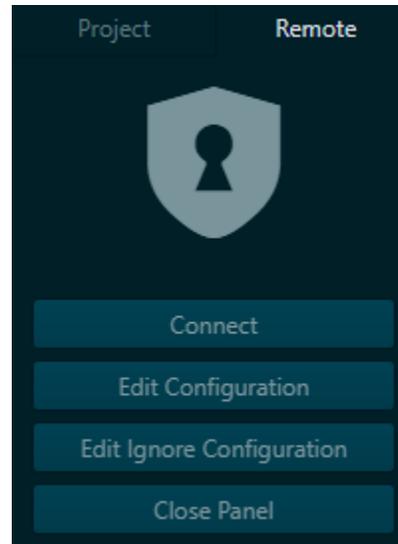
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	id 	bigint(11)			No	Ninguna
2	ph	float			No	0
3	humedad	int(11)			No	0
4	temperatura	int(11)			No	0
5	distancia	int(11)			No	0
6	snr	float			No	0
7	rsi	int(11)			No	0
8	fecha_hora	datetime			No	CURRENT_TIMESTAMP

## FTP

FTP (del inglés, File Transfer Protocol) es un protocolo que permite la transferencia de archivos, se utiliza este protocolo para realizar la carga, descarga y modificación de los archivos php de la plataforma Web desde el ordenador local. Para el proyecto se utiliza *remote-ftp* un paquete para *Atom* que permite habilitar la navegación remota con FTP/FTPS y SFTP, con la opción *Toogle Tree* de la barra de navegación de *Atom* se despliega la pestaña *Remote* para editar la configuración FTP/FTPS.

**Figura 84**

*Pestaña Remote de configuración FTP*



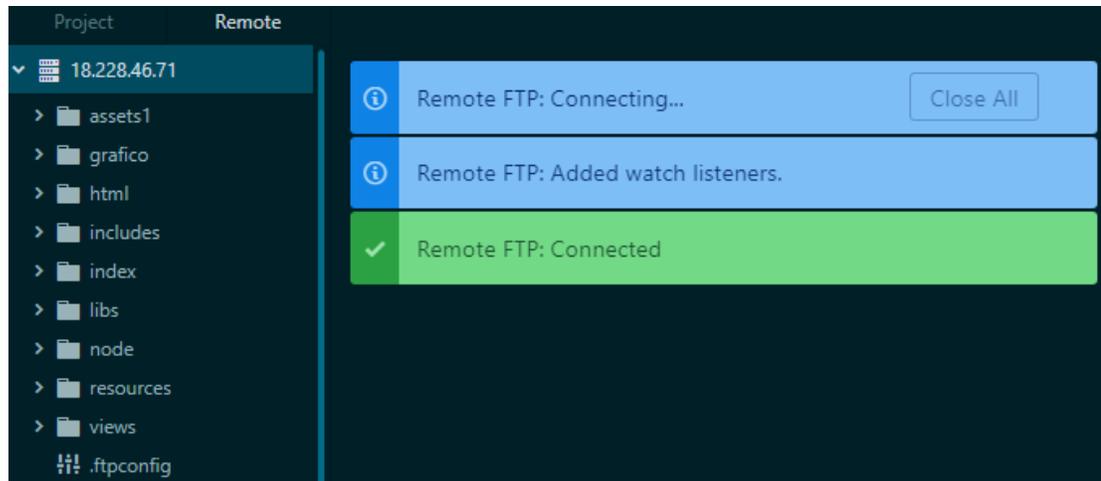
En Remote se elige la opción *Edit Configuration*, de manera automática se crea un fichero con la extensión *ftpconfig*. En el fichero se realiza la configuración que se observa a continuación:

```

1 "protocol": "ftp",
2 "host": "wirelbits.ga", // string -> Nombre del host o dirección IP
3 "port": 21,           //integer -> Puerto del servidor FTP
4 "user": "admin_wirelbits", // string - Nombre de usuario
5 "pass": "xxxxx",      //string -> Contraseña
6 "connTimeout": 10000, //integer -> Tiempo [ms] de espera para
7 conexión
8 "pasvTimeout": 10000, //integer -> Tiempo [ms] de espera para
9 conexión de datos PASV
10 "keepalive": 10000, //integer -> Tiempo [ms] para mantener viva la
    conexión

```

Cabe mencionar que el tiempo *keepalive* es la frecuencia con la que se hace envío de un comando 'dummy' (NOOP) para mantener viva la conexión FTP. Con la opción *Connect* de la pestaña Remote, se accede vía FTP a los archivos de la plataforma IoT alojados en la carpeta *public\_html* del servidor como se observa en la Figura 85.

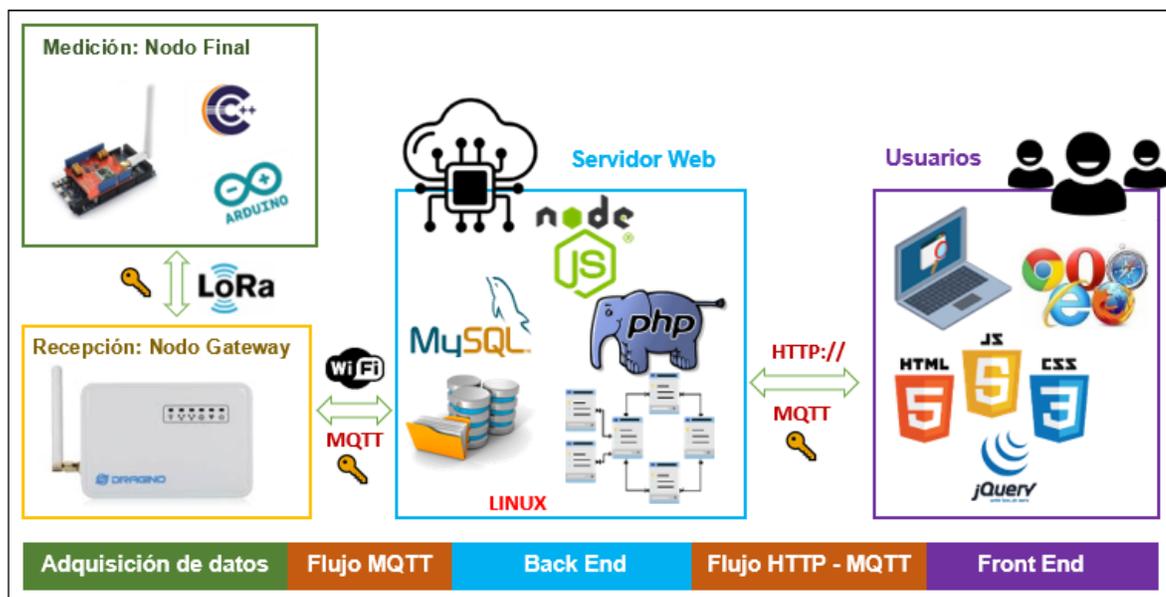
**Figura 85***Conexión FTP*

## Plataforma IoT

La plataforma Web IoT está compuesta de dos partes de programación: Front End que incluye una serie de tecnologías que permiten la interacción de manera directa con el usuario, constituye la parte visual de la plataforma como: íconos, menús desplegables, tipos de letra, colores, efectos, imágenes y movimientos que hacen posible la navegación dentro del sitio, las tecnologías fundamentales para Front End son HTML, JQuery, CSS y JavaScript. Por el contrario, el Back End incluye tecnologías en el servidor que hacen posible la ejecución de procesos y algoritmos para la comunicación, inserción y mantenimiento de información en la base de datos. Las principales tecnologías para desarrollar el Back End son PHP, Node JS, MySQL y Java.

Figura 86

Esquema de tecnologías



*Nota.* La gráfica sintetiza las tecnologías utilizadas en el desarrollo de la plataforma IoT

### **Diseño Front End**

El propósito de la plataforma IoT es proporcionar de un mecanismo de visualización de los datos dinámico, de fácil uso y amigable y. Por medio de la combinación de tecnologías Front End como HTML5, CSS 3 y JavaScript se desarrolla un sitio web con reportes gráficos en tiempo real, consultas de información histórica e inserción de valores de referencia. Se utiliza el framework Bootstrap para garantizar un contenido con una apariencia visual agradable sin importar el tipo de navegador o dispositivo desde el cual se accede y con la biblioteca jQuery se empaqueta el código de una manera más útil y que permite el ahorro de tiempo de diseño. En la Tabla 35, se sintetiza la función de cada una de estas tecnologías en el desarrollo de la plataforma web.

Tabla 35

*Tecnologías utilizadas para el diseño Front End*

Tecnología	Función
<b>HTML 5</b>	Última versión de HTML (del inglés, <i>HyperText Markup Language</i> ), permite crear la estructura de la plataforma Web con soporte de gráficas de alta calidad.
<b>CSS 3</b>	Última versión de CSS (del inglés, <i>Cascading Style Sheets</i> ), lenguaje que permite diseñar y estructurar la parte gráfica de la plataforma desarrollada con HTML. Se usa esta versión porque permite crear efectos de textos, fondos, bordes y transformaciones 2D/3D.
<b>JavaScript</b>	Lenguaje de programación para la web, permite la creación de animaciones y efectos, actualizar, modificar los archivos HTML y CCS.
<b>Bootstrap</b>	Framework más popular para CSS, permite el desarrollo de una plataforma web responsiva y con enfoque mobile-first.
<b>jQuery</b>	jQuery es una librería que permite la optimización con métodos para reducir la escritura de líneas de código JavaScript. Simplifica el uso de AJAX en el proceso de visualización de datos históricos y manejo del DOM.

En la Figura 87, se observa un ejemplo de la estructura HTML5, en la línea 6 se inicia con la importación del framework Bootstrap, en las líneas 7 y 8 se importan las hojas de estilos de CSS3 con el diseño visual de la plataforma, 12 es la importación de la biblioteca jQuery, 13 el complemento JavaScript para Bootstrap y 14 el archivo principal JavaScript con la programación para animaciones y efectos.

## Figura 87

### Estructura básica Front End

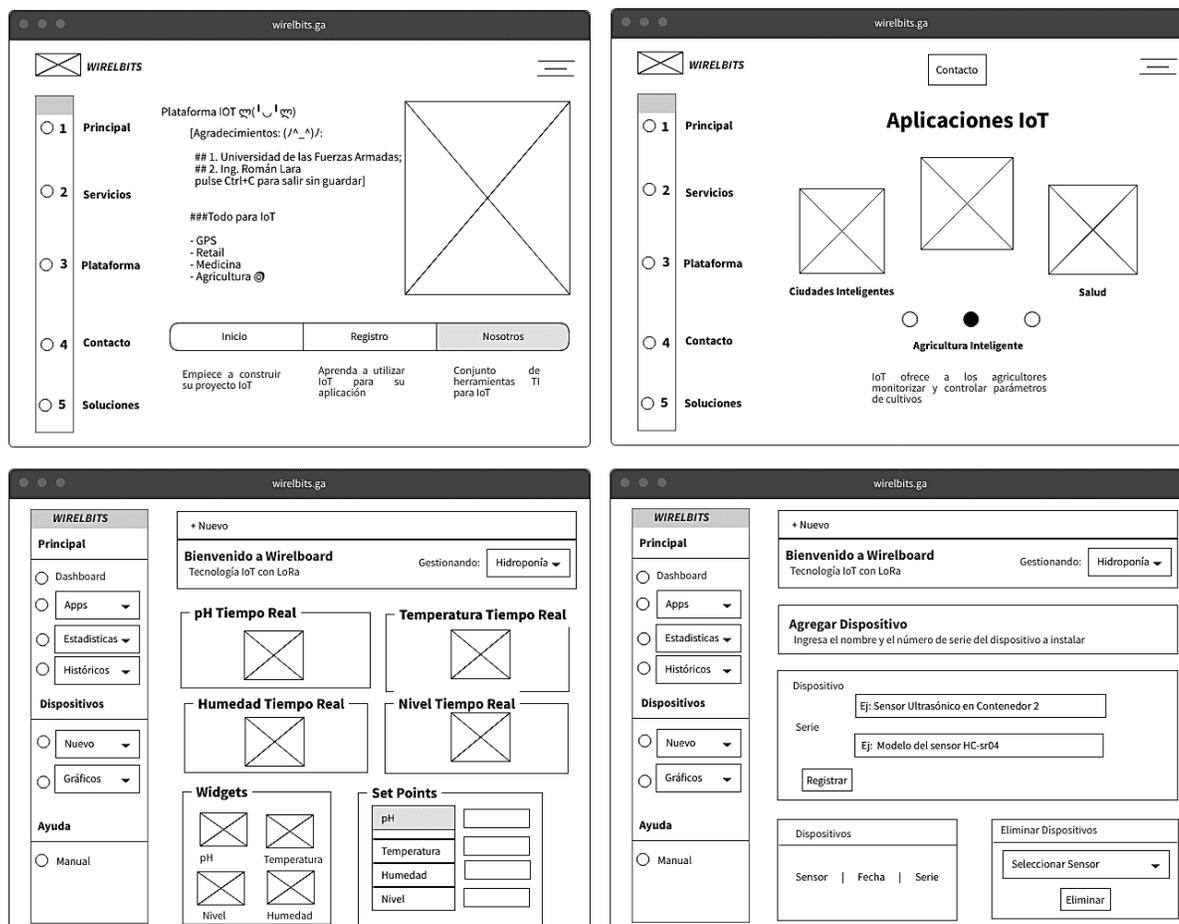
```
extra.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>WirelBits</title>
5  |   <meta charset="utf-8">
6  |   <link rel="stylesheet" href="bootstrap.min.css">
7  |   <link rel="stylesheet" href="index/principal/assets/css/estilos.css">
8  |   <link rel="stylesheet" href="index/principal/assets/css/main.css">
9  </head>
10
11 <body>
12 <script src="jquery.min.js"></script>
13 <script src="bootstrap.min.js"></script>
14 <script src="index/principal/assets/js/functions-min.js"></script>
15 </body>
```

*Nota.* La gráfica presenta un ejemplo de la importación de los archivos requeridos para el diseño gráfico de la plataforma IoT, es importante colocar a jQuery antes de bootstrap.min.js debido a que el complemento JavaScript para Bootstrap necesita previamente la importación de jQuery, al interior de head se coloca la información técnica que entiende el navegador y en body el contenido que aparece en la plataforma web.

En la Figura 88 se presenta el diagrama de la estructura general de la plataforma IoT desarrollada

Figura 88

## Estructura de la plataforma IoT

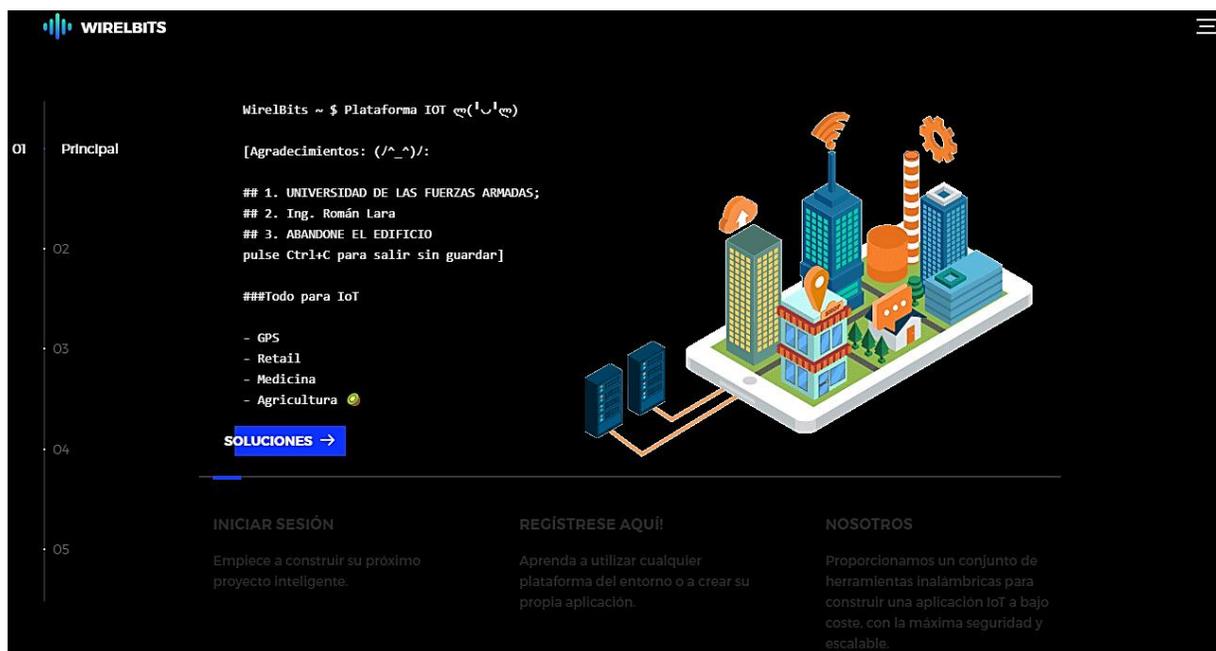


La página principal permite ingresar a la plataforma que contiene la información de todo el proyecto, también dispone de una barra lateral desplegable que permite acceder a 4 secciones: Servicios, Plataforma, Contacto y Soluciones, estas secciones entregan descripciones breves de las soluciones para IoT. Al interior de la plataforma web, se tiene una sección principal que cuenta con reportes gráficos en tiempo real de las variables de monitorización por medio de gadgets, un formulario por el cual los usuarios definen Set Points para la recepción de alertas cuando se sobrepasen esos valores, un componente que realiza cambios de color en la plataforma y en la parte

lateral izquierda un menú con opciones que posibilita el acceso a secciones para la visualización estadística de los datos, consulta de históricos y agregar nuevos dispositivos según las necesidades del usuario, la interfaz principal de la plataforma IoT se presenta en la Figura 89.

**Figura 89**

*Plataforma IoT*



### ***Diseño Back End***

El diseño back end hace referencia a la parte de la plataforma IoT que permite su funcionamiento y al que no pueden acceder los usuarios. Por medio del back end se realiza la gestión, inserción y administración de la base de datos, ejecución de procesos que hacen posible la comunicación, debido a que las peticiones realizadas desde la interfaz de la plataforma requieren de procesamientos realizados por tecnologías como PHP, SQL, JavaScript y Node JS. En la Tabla 36, se sintetiza la función de cada una de estas tecnologías en la programación de la plataforma web.

**Tabla 36**

*Tecnologías utilizadas para la programación Back End*

<b>Tecnología</b>	<b>Función</b>
<b>PHP</b>	Lenguaje de programación que hace posible la lectura de la base de datos desde la interfaz de la plataforma, permite la generación de los reportes gráficos y estadísticos.
<b>SQL</b>	Lenguaje de consulta estructurado que permite la administración de la base de datos y realización de múltiples operaciones con los datos que almacena.
<b>JavaScript</b>	Lenguaje en el que se base Node JS permite hacer cálculos, manipulación e inserción de los datos transmitidos por los sensores.
<b>Node JS</b>	Entorno de ejecución, establecido como el punto clave de la plataforma IoT ya que hace posible la inserción y almacenamiento de la información en la base de datos, así como también permite la emisión de alertas vía SMS.

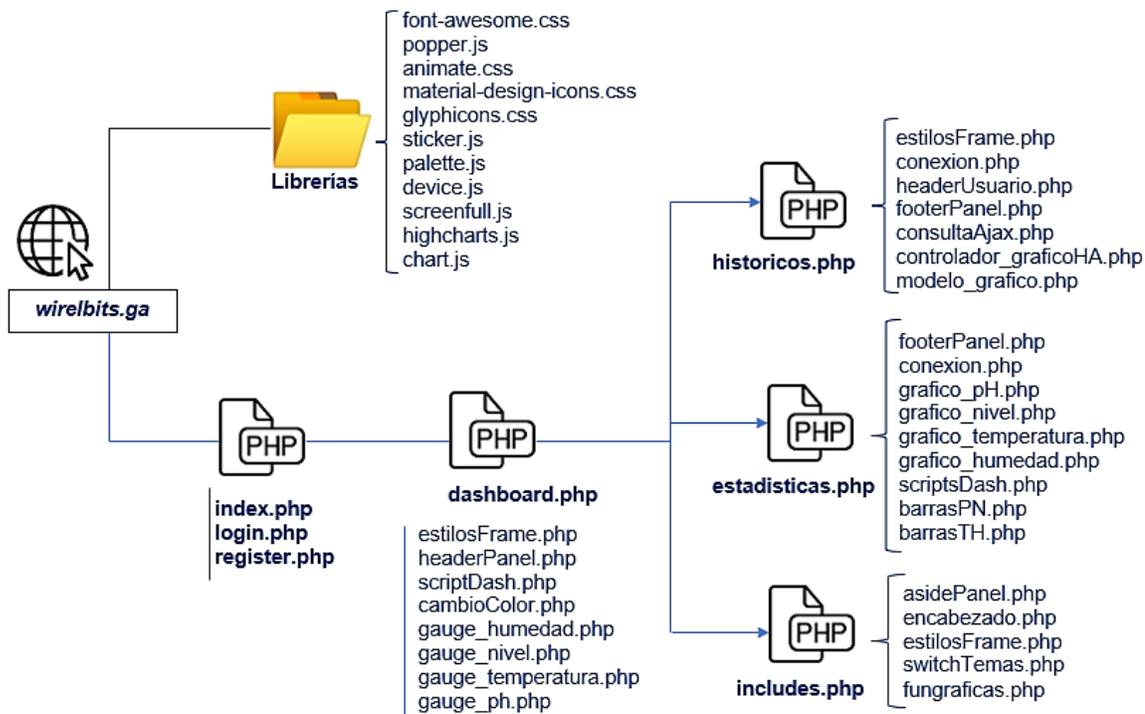
*Nota.* Node JS, presenta afinidad con múltiples tecnologías IoT, en el repositorio NPM existen más de 80 paquetes para Arduino.

### ***Archivos de la Plataforma IoT***

La plataforma web incluye un conjunto de archivos con códigos y librerías que operan de manera conjunta para formar un único sistema, en la Figura 90, se realiza el diagrama de jerarquías donde se observa los diferentes archivos que hacen posible el funcionamiento de la interfaz, así como también de procesos en el servidor para la visualización estadística de los datos de las variables físicas.

Figura 90

Diagrama de archivos de la Plataforma IoT

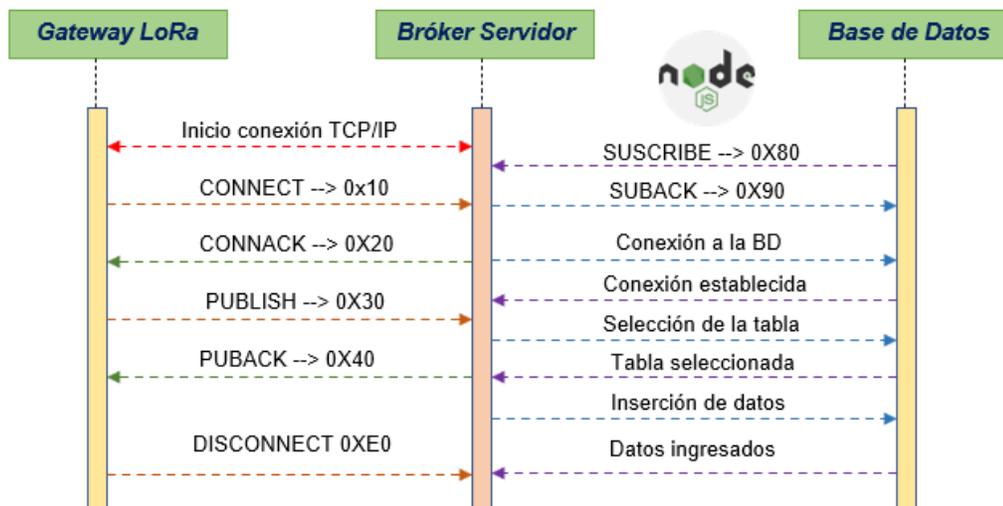


### Comunicación Gateway LoRa-Bróker-BD

Luego de crear las BD de la Figura 91, el cliente *Gateway* realiza una conexión de tipo TCP/Ip con el bróker del servidor, para ello se hace uso de los puertos 8093 y 8094 cuando opera sobre TLS. La conexión permanece abierta hasta cuando el *Gateway* la finaliza, el proceso de comunicación entre la pasarela y el bróker se observa a continuación:

**Figura 91**

Esquema de transmisión y almacenamiento de datos en la BD



### Configuración de Node JS

Después de realizar la conexión vía SSH a la instancia, se efectúa la instalación de Node JS con la herramienta APT de Linux, para ello se requiere ser un usuario *root* y desde la línea de comandos escribir *apt install nodejs*. También es necesario instalar el gestor de paquetes *npm*, de la misma manera en la terminal se escribe *apt install npm*. Una vez realizado la instalación de ambas herramientas se realiza la verificación de las versiones con los comandos de la Figura 92.

**Figura 92**

Versiones de Node JS y NPM

```
root@wirelbits:/# node --version
v8.10.0
root@wirelbits:/# npm -v
3.5.2
```

Para la inserción de los datos se necesita del controlador *node.js* para MySQL y la biblioteca cliente *MQTT.js* para el protocolo MQTT y el navegador. La instalación de

ambos componentes se realiza por medio de npm en la carpeta que contiene todos los archivos de la plataforma IoT, como se observa en la Figura 93.

**Figura 93**

*Instalación de componentes para Node JS*

```
root@wirelbits:/home/admin/web/wirelbits.ga/public_html/node# npm install mysql
transmission_lora@1.0.0 /home/admin/web/wirelbits.ga/public_html/node
└─ mysql@2.18.1 extraneous
root@wirelbits:/home/admin/web/wirelbits.ga/public_html/node# npm install mqtt
└─
WARN engine mqtt@4.2.6: wanted: {"node": ">=10.0.0"} (current: {"node": "8.10.0", "npm": "3.5.2"})
```

Node JS permite la inserción en tiempo real de los datos, para ello en el archivo *index.js* se inicializa las variables y se definen las credenciales para MySQL y MQTT como se observa a continuación:

```
1 var mysql = require('mysql');
2 var mqtt = require('mqtt');
3 //CREDENCIALES MYSQL
4 var con = mysql.createConnection({
5   host: "wirelbits.ga",
6   user: "admin_wirelbits",
7   password: "xxxxxx",
8   database: "admin_lorawan"
9 });
10 //CREDENCIALES MQTT
11 var options = {
12   port: 1883,
13   host: 'wirelbits.ga',
14   clientId: Math.round(Math.random() * (0-10000) * -1),
15   username: 'web_client',
16   password: 'xxxxxx',
17   keepalive: 60,
18   reconnectPeriod: 1000,
19   protocolId: 'MQIsdp',
20   protocolVersion: 3,
21   clean: true,
22   encoding: 'utf8'
23 };
```

Luego de realizar la conexión al bróker, se suscribe al tópic y con el comodín "#" se reciben todos los datos transmitidos por el *Gateway* con el siguiente código:

```

1 client.on('connect', function () {
2 console.log("Conexión MQTT Exitosa!");
3 client.subscribe('sensores/#', function (err) {
4 console.log("Subscripción exitosa!")
5 });
6 })

```

Cuando se recibe un mensaje, se realiza el proceso de decodificación para ello primero se convierte a *Strings* y con el método `split ()` se divide el objeto en un vector que permite separarlo en subcadenas por medio de un parámetro "separador". Se realiza la consulta para realizar la inserción, un ejemplo general del código es el siguiente:

```

1 client.on('message', function (topic, message) {
2   if (topic == "sensores"){
3     var msg = message.toString();
4     var sp = msg.split(",");
5     var phS = sp[1];
6     var humedadS = sp[2];
7     //Consulta para insertar....
8     var query = "INSERT INTO `admin_lorawan`.`tabla_sensor` (`ph`,
9 `humedad`) VALUES (" + phS + "," + humedadS + ");";
10    con.query(query, function (err, result, fields) {
11      if (err) throw err;
12      console.log("Fila insertada correctamente");
13    });
14  }
15 });

```

Para la automatización de la bomba se parte del criterio del algoritmo de la Figura 63, al estar la bomba inicialmente encendida se envía la instrucción de apagado una sola vez luego de haber transcurridos 5 minutos con el uso de la función `setTimeout ()` y bajo el tópic "sensorp". `setInterval ()` permite la ejecución de la función de automatización de encendido de manera repetitiva cada 15 minutos, el código utilizado es el siguiente:

```

1 function encender(){
2   client.publish('sensorp', 'on', (error) => {
3     console.log(error || 'Encendiendo bomba...!!!')
4   })
5 }
6
7 function apagar(){
8   client.publish('sensorp', 'off', (error) => {
9     console.log(error || 'Apagando bomba...!!!')
10  })
11 }
12 setTimeout(apagar, 300000);
13 setInterval(encender, 900000);

```

Por medio de *Node.js* también es posible realizar la integración con otras plataformas. El siguiente código realiza la transferencia de datos hacia la plataforma *IoT ThingSpeak* con la finalidad de visualizar y analizar la información en tiempo real en la nube.

```

1 var ThingSpeakClient = require('thingspeakclient');
2 var client = new ThingSpeakClient();
3 var WriteKey = 'EATLI1NNXUGHJ1HQ';
4 var channelID = 72366;
5 client.attachChannel(channelID, { writeKey:'WriteKey'},
6 callBackThingspeak);
7 client.updateChannel(channelID, {field1: +pH+, field2: +nivel+},
8 function(err, resp) {
9   if (!err && resp > 0) {
10     console.log('Número de entrada: ' + resp);}
11   else {
12     console.log(err);}
13 function callBackThingspeak(err, resp){
14   if (!err && resp > 0) {
15     console.log('La respuesta es: ' + resp);}
16   else {
17     console.log(err);}

```

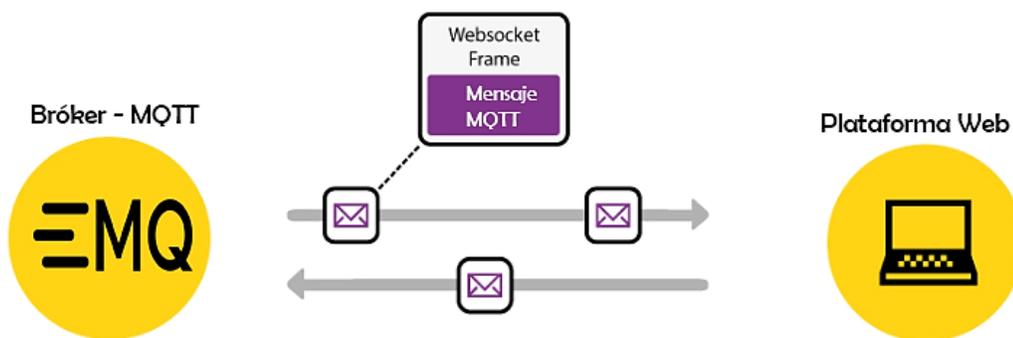
### **Comunicación Bróker-Plataforma IoT**

Para la comunicación entre la plataforma IoT y el bróker, se hace uso de Websockets para el envío y recepción de datos de manera directa y bidireccional entre el navegador web y el servidor. En MQTT sobre WebSockets, los mensajes se

encapsulan por medio de una o más tramas WebSocket y son transferidos a través de la red. EMQ es un bróker con la capacidad de manejar WebSockets como se observa en la Figura 94.

**Figura 94**

*MQTT y WebSockets*



Para presentar los datos en tiempo real por medio de los gadgets, de la misma forma se establecen las credenciales MQTT, se inicializa la conexión vía *WebSocket*, por medio de la consola del navegador se verifica que la conexión MQTT sea exitosa. Para el proceso de decodificación se crea una función que recibe los cuatro valores de sensado y los parámetros de QoS como se observa a continuación:

```

1 const WebSocket_URL = 'wss://wirelbits.ga:8094/mqtt'
2 const client = mqtt.connect(WebSocket_URL, options)
3 client.on('connect', () => {
4   console.log('Mqtt conectado por WSS! Exito!')
5 })
6 client.on('message', (topic, message) => {
7   process_msg(topic, message);
8 })

```

Se procede a crear una función que recibe como parámetros el tópic y el mensaje transmitido por el bróker, de la misma manera se realiza la decodificación con los métodos anteriormente definidos como se observa a continuación:

```

1 function process_msg(topic, message) {

```

```

2  if (topic == "sensores"){
3    var msg = message.toString();
4    var sp = msg.split(",");
5    var temp = sp[0];
6    var nivel = sp[1];
7    var ph = sp[2];
8    var hum = sp[3];
9    var snr1= sp[4];
10   var rssi1 = sp[5];
11   update_values(temp,nivel,ph,hum,snr1,rssi1);
12   }
13 }

```

Para obtener el contenido de cada variable se hace uso de *.html()* y con el selector *\$* de jQuery se define donde se visualizan los datos.

```

1  function update_values(temp,nivel,ph,hum,snr1,rssi1) {
2    $("#display_temp").html(temp);
3    $("#display_hum").html(hum);
4    $("#display_nivel").html(nivel);
5    $("#display_ph").html(ph);
6    $("#display_snr").html(snr1);
7    $("#display_rssi").html(rssi1);
8  }

```

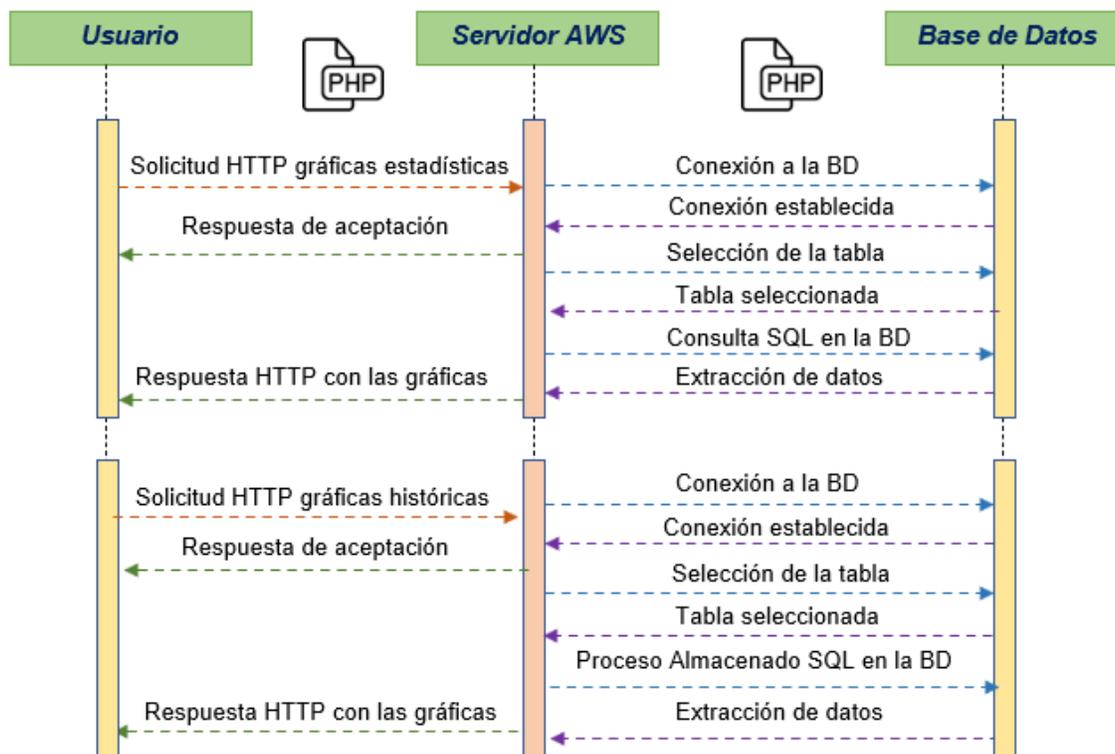
### **Comunicación Plataforma IoT – BD**

La página principal de la plataforma (*index.php*) se encarga realiza de manera dinámica la primera interacción con los usuarios y permite el acceso al archivo *dashboard.php* que contiene la información en tiempo real. En el dashboard se tiene una barra lateral con pestañas para acceder a la consulta de valores históricos desde la base de datos y reportes estadísticos. Los archivos *estadisticas.php* permiten la generación de las gráficas mediante la combinación de los archivos PHP de la Figura 97. Las imágenes disponen de la funcionalidad de descarga en formatos *jpg* y *png*.

El diagrama de la Figura 95, define la secuencia de comunicación para la obtención de las gráficas e históricos de la base de datos.

Figura 95

Diagrama comunicación entre usuario y servidor



Para la generación de las gráficas se hace uso de *Highcharts JS* que es una biblioteca eficiente para la creación de gráficos dinámicos y complejos con JavaScript, para su uso se realiza la importación de los archivos, por medio del atributo `src` se especifica la carpeta de ubicación como se observa a continuación:

```

1 <script src="../../../resources/code/highcharts.js"></script>
2 <script src="../../../resources/code/modules/series-label.js"></script>
3 <script src="../../../resources/code/modules/exporting.js"></script>
4 <script src="../../../resources/code/modules/export-data.js"></script>
5 <script src="../../../resources/code/modules/accessibility.js"></script>
  
```

La conexión hacia la base de datos es necesaria para la generación de los reportes gráficos, con el siguiente código se realiza la creación de funciones para inicializar, conectar y finalizar la conexión:

```

1 public function __construct() {
2     $this->servidor = "wirelbits.ga";
3     $this->usuario = "admin_wirelbits";
4     $this->contrasena = "xxxxxxx";
5     $this->basedatos = "admin_lorawan";
6 }
7 function conectar() {
8     $this->conexion = new mysqli($this->servidor,$this->usuario,$this->
9 >contrasena,$this->basedatos);
10    $this->conexion->set_charset("utf8");
11 }
12 function cerrar() {
13    $this->conexion->close();
14 }

```

Por medio de la función *require\_once()* se incluye al fichero de conexión para la ejecución del script.

```

1 <?php
2 require_once("conexion.php");
3 ?>

```

Se desarrolla una función donde se define las propiedades básicas como: largo, ancho y tipo de gráfica, colores, estilos, tamaños de letra, se realiza la consulta hacia la base de datos y con el selector de jQuery *#id* se crea un contenedor para almacenar las gráficas. Por medio de *mysqli\_fetch\_array* se obtiene una fila de resultados como un array de tipo asociativo y con *while* se recorre al mismo, una síntesis del código se observa a continuación:

```

1 $(function () {
2     $('#container').highcharts({
3     chart: {
4     spacingBottom: 1,
5     spacingTop: 5,
6     spacingLeft: 5,
7     spacingRight: 5,
8     width: 400,
9     height: 300,

```

```

10  type: 'gauge',
11  },
12  series: [{
13  name: 'Nivel de pH',
14  data: [
15  <?php
16  $query = " select ph from tabla_sensor order by id desc limit 1 ";
17  $resultados = mysqli_query($connection, $query);
18  while($rows = mysqli_fetch_array($resultados)){
19  ?>
20  <?php
21  echo $rows["ph"]?>,
22  <?php
23  }
24  ?>],
25  tooltip: {
26  valueSuffix: ' pH'
27  }
28  }]

```

Para actualizar la información en tiempo real, se hace uso del método *reload ()* encargado de recargar el documento actual, función similar al del botón de recarga del navegador.

```

1 <script type="text/javascript">
2 setTimeout('document.location.reload()',5000)
3 </script>

```

La consulta de datos históricos en la base de datos es posible gracias al método AJAX, que hace posible intercambiar datos con el servidor sin la necesidad de recargar la página web. Se crea una función donde se obtiene el valor de las fechas para realizar la consulta y se concatena en variables, como se observa en el siguiente código:

```

1 var fechaI=document.getElementById('fechaI').value;
2 var horaI=document.getElementById('startT').value;
3 var fechaF=document.getElementById('finF').value;
4 var horaF=document.getElementById('endF').value;
5 var fechaTI=fechaI+" "+horaI;
6 var fechaTF=fechaF+" "+horaF;

```

El método `$.ajax()` se configura por medio de un objeto, el mismo que almacena múltiples operaciones que jQuery requiere para realizar una petición. Al interior del método se define la URL de hacia dónde se realiza la petición, se establece el tipo de petición y la información que se envía para la consulta, la configuración se observa a continuación:

```

1 $.ajax({
2   url:'includes/chartjs/controlador_graficoHF.php',
3   type:'POST',
4   data:
5     {
6       inicioA:fechaTI,
7       finF:fechaTF
8     }

```

En el archivo `controlador_grafico.php` se reciben los datos por medio del método POST, se crea una función que toma como parámetros las fechas de búsqueda y se realiza la codificación del array asociativo en un objeto JSON con el siguiente código:

```

1 <?php
2 require 'modelo_grafico.php';
3
4 $fechaTI = $_POST['inicioA'];
5 $fechaTF = $_POST['finF'];
6
7 $MG1 = new Modelo_Grafico();
8 $consulta = $MG1 -> traerDatosConsultaHF($fechaTI,$fechaTF);
9 echo json_encode($consulta);
10 ?>

```

Con la sentencia `require` se realiza la inclusión del archivo `modelo_grafico.php`, donde se lleva a cabo la conexión y consulta de históricos hacia la base de datos mediante un procedimiento almacenado:

```

1 function traerDatosConsultaHF($fechaTI,$fechaTF) {
2   $sql = "CALL SP_DATOSCONSULTAHA('$fechaTI','$fechaTF')";
3   $arreglo = array();
4   if ($consulta = $this->conexion->conexion->query($sql)) {
5     while ($consulta_VU = mysqli_fetch_array($consulta)) {

```

```

6     $arreglo[] = $consulta_VU;
7     }
8     return $arreglo;
9     $this->conexion->cerrar();
10    }
11 }

```

Cabe mencionar que la BD a la que se dirige la sentencia CALL tiene soporte para el llamado de procesos almacenados. En la Figura 96, se observa un ejemplo del proceso de creación de uno de los procedimientos almacenados.

**Figura 96**

*Proceso almacenado para consulta de históricos*

The screenshot shows a 'Details' window for a stored procedure. The 'Routine name' is 'SP\_DATOSCONSULTAHF' and the 'Type' is 'PROCEDURE'. Below this, a table lists the parameters:

Direction	Name	Type	Length/Values	Options
IN	FECHAINICIO	VARCHAR	30	Charset
IN	FECHAFIN	VARCHAR	30	Charset

Below the parameters table, there is an 'Add parameter' button and a text area containing the following SQL query:

```

1 select `humedad`, `fecha_hora` from tabla_sensor WHERE `fecha_hora` BETWEEN FECHAINICIO AND FECHAFIN

```

En la Figura 97, se presenta todos los procedimientos almacenados creados y que hacen posible la consulta de datos históricos.

Figura 97

Procedimientos almacenados

Routines		Name	Action	Type	Returns
<input type="checkbox"/>		SP_DATOSCONSULTAHA	Edit  Execute  Export  Drop	PROCEDURE	
<input type="checkbox"/>		SP_DATOSCONSULTAHF	Edit  Execute  Export  Drop	PROCEDURE	
<input type="checkbox"/>		SP_DATOSCONSULTANA	Edit  Execute  Export  Drop	PROCEDURE	
<input type="checkbox"/>		SP_DATOSCONSULTANF	Edit  Execute  Export  Drop	PROCEDURE	
<input type="checkbox"/>		SP_DATOSCONSULTATA	Edit  Execute  Export  Drop	PROCEDURE	
<input type="checkbox"/>		SP_DATOSCONSULTATF	Edit  Execute  Export  Drop	PROCEDURE	

Si se tiene una petición que resulta satisfactoria, con *JSON.parse()* los datos son convertidos a un objeto JavaScript, debido a que la información recibida desde el servidor siempre es una cadena. Con un *for* se recorre el arreglo, con el método *push()* se añaden los datos en nuevos vectores y se crea una función que recibe los siguientes parámetros para la generación de las gráficas históricas:

```

1 done(function(resp) {
2   var data=JSON.parse(resp) ;
3   var fechas=[];
4   var datos=[];
5   var colores=[];
6   for(var i=0;i<data.length;i++){
7     fechas.push(data[i][1]);
8     datos.push(data[i][0]);
9     colores.push(colorRGB());
10  }
11  crearGraficoAcumuladoH(fechas,datos,'scatter',colores);
12 })

```

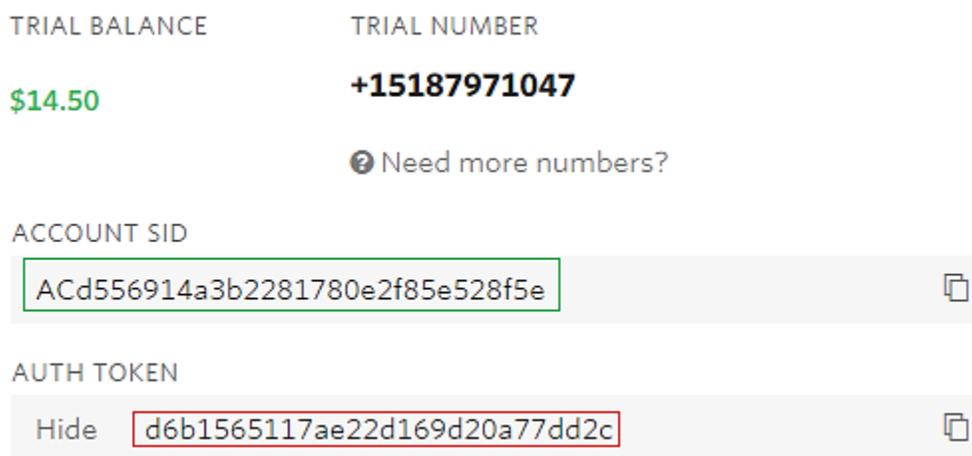
### Configuración de la plataforma para el envío de alertas

Para el envío de alertas se hace uso de la plataforma Twilio, que hace posible el desarrollo de aplicaciones para la oferta de múltiples servicios de comunicaciones como: envío y recepción de llamadas, mensajería de texto SMS, video, servicio de

correo electrónico, WhatsApp y SIP por medio de *APIs* (del inglés, *Application Programming Interface*). Luego de realizar el registro en la plataforma, la Figura 98 expone los parámetros iniciales para la configuración del servicio de mensajería.

### Figura 98

*Información del servicio*



The image shows a screenshot of a Twilio account dashboard. It displays the following information:

- TRIAL BALANCE:** \$14.50
- TRIAL NUMBER:** +15187971047
- Need more numbers?** (with a help icon)
- ACCOUNT SID:** ACd556914a3b2281780e2f85e528f5e
- AUTH TOKEN:** d6b1565117ae22d169d20a77dd2c

*Nota.* Las credenciales de Twilio se encuentran en el panel de control de la consola, en la cual también se puede obtener números de envío y otras funcionalidades.

Antes del inicio de cualquier proyecto con *Node.js* se requiere ejecutar *npm init* con la finalidad de crear un nuevo archivo *package.json* como se observa a continuación:

## Figura 99

### Manejador de dependencias



```

node > {} package.json > ...
1  {
2  |   "name": "transmision_lora",
3  |   "version": "1.0.0",
4  |   "description": "Almacenamiento en BD",
5  |   "main": "index.js",
6  |   "author": "Andres Loya",
7  |   "license": "ISC",
8  |   "dependencies": {
9  |     "twilio": "^3.62.0"
10 | }

```

*Nota.* `package.json` es un archivo que incluye todos los metadatos del proyecto como: el nombre, la versión, licencia, las dependencias y los scripts.

Para comenzar la creación del servicio de alertas con SMS se requiere instalar el nodo Twilio por medio de npm, como se presenta en la Figura 100.

## Figura 100

### Instalación de nodo Twilio



```

root@wirelbits:/home/admin/web/wirelbits.ga/public_html/node# npm install twilio
transmision_lora@1.0.0 /home/admin/web/wirelbits.ga/public_html/node
├─ mqtt@4.2.6 extraneous
├─ mysql@2.18.1 extraneous
├─ twilio@3.67.0
├─ axios@0.21.1
├─ follow-redirects@1.14.2
├─ dayjs@1.10.6
└─ https-proxy-agent@5.0.0

```

Luego de la instalación del nodo, en el archivo `index.js` se registra los parámetros: `accountSid` y `authToken`, se crea una función donde se definen los números de origen, destino y el mensaje SMS a enviar, como se observa a continuación:

```

1 const twilio =require('twilio');
2 const accountSid='AC0b02d0e75f4f090647a0bb5151f8f1ac';
3 const authToken='e69a0d595b8ae5b89dc3c220d1cac11c';
4 const client= new twilio(accountSid,authToken);
5
6 const createSMS1 = () => {
7   client.messages.create({
8     body:'Hola, ;;revisa el sensor de pH urgentemente!!!',
9     to:'+59309832830xx',
10    from:'+15187971047'
11  }).then((message) => console.log('SMS1 ===> ENVIADO'));
12 }

```

Se realiza una consulta a la base de datos de los sensores y set Points con la finalidad de comparar ambas variables y emitir las alertas con el siguiente código:

```

1 con.query("select ph from tabla_sensor order by id desc limit 1",
2 function (err, result) {
3   if (err) throw err;
4   pHS = Object.values(JSON.parse(JSON.stringify(result)));
5   });
6 con.query("select pH from set_points order by id desc limit 1",
7 function (err, result) {
8   if (err) throw err;
9   pHM= Object.values(JSON.parse(JSON.stringify(result)));
10  console.log(tempS);
11  });
12 if (pHS<pHM) {
13   console.log("pH Correcto!")
14 }else{
15   createSMS1 ();
16 }

```

### Implementación de la red de sensores

Una vez realizado el desarrollo y programación de la plataforma IoT para la recepción, almacenamiento y visualización de los datos, se procede a realizar la implementación de la red de sensores previamente calibrados.

#### *Descripción del escenario*

El escenario en el cual se realiza la implementación es en el sistema hidropónico localizado al interior de un invernadero, una estructura cubierta que permite el cultivo de

diversas plantas en un ambiente que puede ser controlado, cabe mencionar que las variables ambientales del invernadero están sujetas a las variaciones climáticas del lugar de ubicación. El invernadero es de tipo túnel, formado por estructura metálica, cubierto de plástico transparente, mallas laterales para permitir una adecuada ventilación, mayor resistencia al viento, niveles altos de transmisión de la luz y evacuación sencilla del agua de lluvia. Las dimensiones del invernadero se observan en la Tabla 37.

**Tabla 37**

*Dimensiones del invernadero*

<b>Parámetro</b>	<b>Dimensión (m)</b>
<b>Alto</b>	2
<b>Longitud de cuerda</b>	8
<b>Flecha para arcos</b>	1.25
<b>Ancho</b>	6

En la Figura 101, se comprueba que el invernadero presenta una curvatura en el techo y sus características permiten garantizar el cuidado del nodo final a agentes externos ambientales y el crecimiento óptimo de las plantas en el sistema hidropónico diseñado.

**Figura 101**

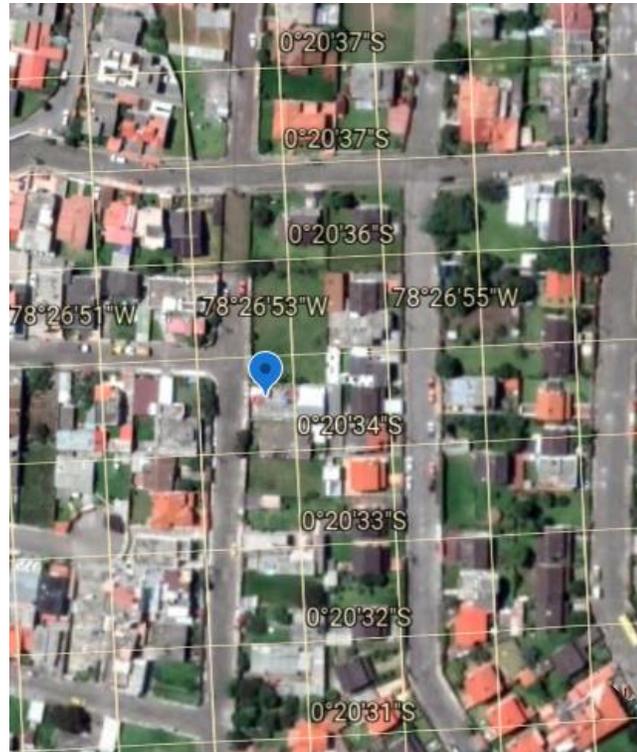
*Vista externa del invernadero*

***Localización geográfica***

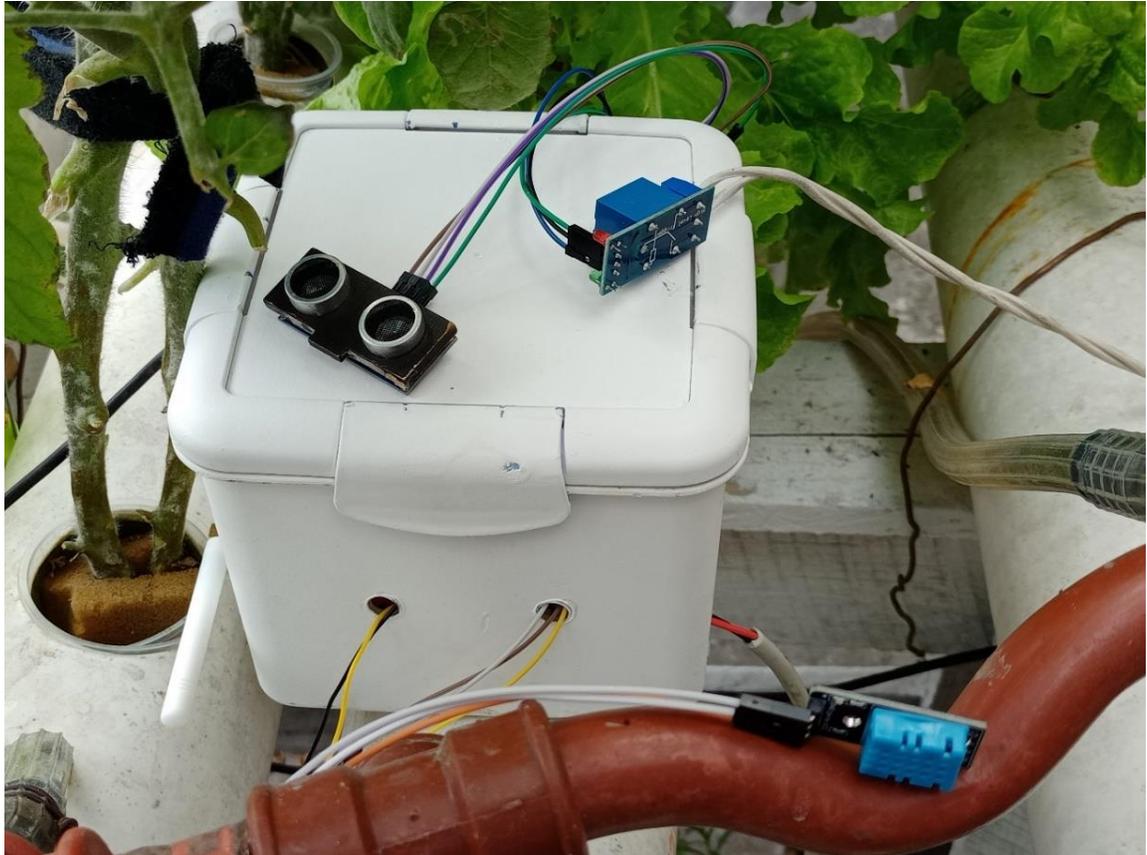
El escenario donde se efectúa la implementación de la red de monitorización se encuentra ubicado en el sector de Sangolquí en el cantón Rumiñahui. Por medio del software Google Earth Pro se puede visualizar de manera satelital el lugar exacto de implementación, como se aprecia en la Figura 102.

**Figura 102**

*Localización geográfica*

***Implementación del nodo final***

En base al circuito esquemático y diseño de la caja expuesto en las Figura 48 y Figura 65 respectivamente, la Figura 103 presenta el nodo Final desplegado en el sistema hidropónico, compuesto de la caja de material PVC cuya función es proteger a los sensores de agentes ambientales como el agua, polvo y otros parámetros que puede afectar el funcionamiento del transceptor LoRa, los sensores, microcontrolador y convertidor de voltaje.

**Figura 103***Nodo Final*

Con la finalidad de habilitar la tecnología LoRa e iniciar el proceso de monitorización, sujeto a las consideraciones de altura de antena de Tx y Rx establecido en la Tabla 32, en la Figura 104 se aprecia la implementación en el sistema hidropónico.

**Figura 104***Implementación del Nodo Final*

## Capítulo V

### Pruebas y Resultados

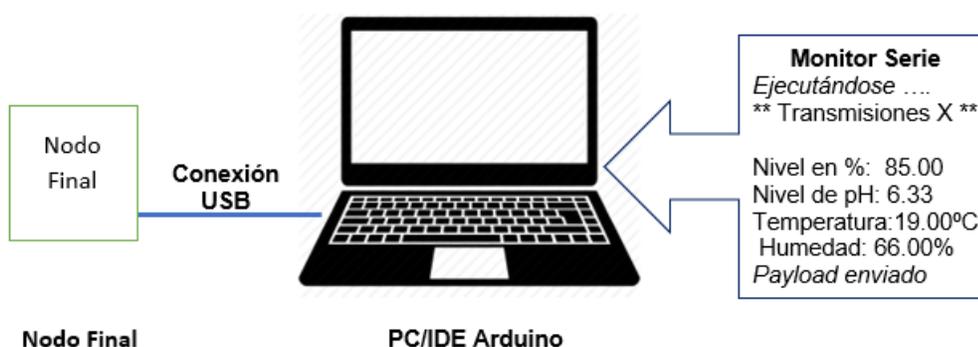
#### Protocolo de pruebas

##### Adquisición de datos

La primera prueba se aplica en base al escenario de prueba de la Figura 105 y se efectúa con la finalidad de verificar que exista un óptimo funcionamiento del nodo final en el proceso de monitorización, antes de la recolección final de los datos de las variables físicas.

#### Figura 105

*Escenario adquisición de datos*



En primer lugar, se enciende el nodo final y se realiza la conexión del transceptor LoRa a la computadora por medio de un cable USB. Se inicia el IDE de Arduino, se ejecuta el programa de transmisión con LoRa y se habilita el monitor serie para observar los valores adquiridos por los cuatro sensores. Es importante considerar que previamente se realizó el proceso de calibración y compensación por medio de software como se presentó en la sección dedicada al diseño del nodo final. En la Figura 106 se observa la adquisición de las variables físicas en el monitor serie de Arduino, donde también existe un contador de las transmisiones correctas efectuadas.

**Figura 106***Adquisición de datos*

```

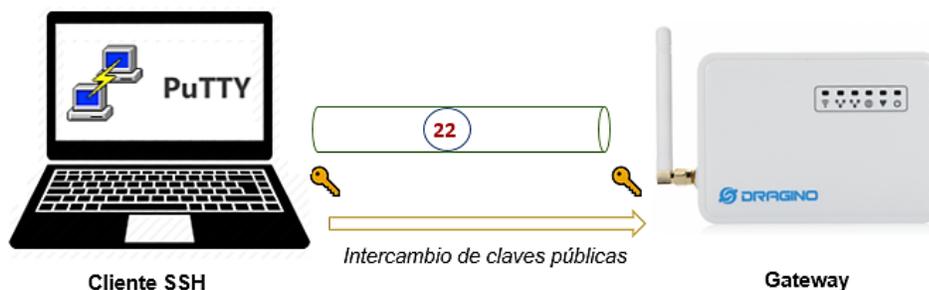
COM4
##### Iniciando Comunicación... #####
##### Comunicación via LoRa exitosa!!... #####
##### Transmisiones Correctas:1 #####
Los datos se envían...

Nivel de Solución en %: 85.00
Nivel de pH: 6.33
Las mediciones de Temperatura y Humedad:
[18.00°C,85.00%]
SNR: 10.75
RSSI: -94
Payload enviado correctamente!!

```

**Recepción de datos en el Gateway**

La Figura 107 muestra el escenario de prueba para la verificación de comunicación entre el Gateway-Nodo Final y corroborar la recepción correcta de los datos transmitidos por el transceptor LoRa.

**Figura 107***Esquema de verificación de recepción*

Para esta prueba se emplea el software Putty, una vez realizada la conexión vía SSH se realiza el *login* con el usuario y contraseña definidos en la Figura 67 y desde la consola se escribe el comando que se observa en la Figura 108.

## Figura 108

Conexión vía SSH al Gateway

```

DRAGINO
W i F i , L i n u x , M C U , E m b e d d e d

OpenWRT 18.06
Version: Dragino-v2 lgw-5.4.1609858793
Build Tue Jan  5 22:59:53 CST 2021

www.dragino.com
-----
root@dragino-1f6f78:~# logread -f

```

Por medio de los logs es posible verificar y validar que existe una transmisión precisa de los datos, como se observa a continuación:

## Figura 109

Paquete de datos recibido

```

daemon.info lg01_pkt_fwd[1539]:
daemon.info lg01_pkt_fwd[1539]: [RXTX- Receive (HEX):3c343536373e31382e302c38352e302c362e33332c2038362e302c31302e37352c2d3934
daemon.info mqtt_process.sh[1608]: /var/iot/channels/ CREATE 4567
user.notice root: [IoT.MQTT]: Find Match Entry for 4567
user.notice root: [IoT.MQTT]: DECODER 4567

```

El payload recibido presenta la estructura definida en la *Tabla 38*, para el proceso de decodificación se convierte de hexadecimal a carácter basado en la tabla ASCII.

## Tabla 38

Ejemplo de decodificación del Payload

---

<b>Payload (HEX)</b>	<b>3c343536373e31382e302c38352e302c362e33332c2038372e302c31302e37352c2d3934</b>
----------------------	---

---

Canal	Tipo	Valor (Char)
	Temperatura: 31382e30	18.0
3c343536373e	Humedad: 38352e30	85.0
	pH: 362e3333	6.33
	Nivel: 38372e30	87.0
<4567>	Snr: 31302e3735	10.75
	Rssi: 2d3934	-94

*Nota.* En el payload se incluye comas y espacios representadas por los valores 2c y 20 respectivamente, esto se hace con la finalidad de facilitar el proceso de decodificación.

### Conectividad

El *Gateway* LoRa constituye el elemento central en el proceso de comunicación de los datos transmitidos con LoRa hacia Internet, es por ello que en esta prueba se realiza la verificación de conectividad. Por medio del Software *Advanced IP Scanner* se verifica los dispositivos enlazados a la red WiFi, sus respectivas direcciones IP y MAC, como se aprecia en la Figura 110.

### Figura 110

*Red WiFi*

IP	Fabricante	Dirección MAC
192.168.100.1	HUAWEI TECHNOLO...	D4:F9:A1:BA:49:16
192.168.100.107	Dragino Technology ...	A8:40:41:1F:6F:7A

Desde la consola del computador conectado a la misma red del *Gateway*, se realiza una petición ICMP por medio del comando ping hacia la dirección IP o DNS del servidor y al *Gateway*. Luego de realizar un ping al *Gateway* LoRa se obtuvo un total de

15 paquetes entre enviados y recibidos, dando un total de 0 paquetes perdidos y los tiempos que se observan en la Figura 111.

### Figura 111

#### *Petición ICMP al Gateway LoRa*

```
C:\Users\ADMIN>ping 192.168.100.107 -t

Haciendo ping a 192.168.100.107 con 32 bytes de datos:
Respuesta desde 192.168.100.107: bytes=32 tiempo=4ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=13ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=61ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=6ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=4ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=11ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=14ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=16ms TTL=64
Respuesta desde 192.168.100.107: bytes=32 tiempo=1ms TTL=64

Estadísticas de ping para 192.168.100.107:
    Paquetes: enviados = 15, recibidos = 15, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 1ms, Máximo = 61ms, Media = 9ms
```

Con la finalidad de establecer la latencia presente en el enlace inalámbrico se realiza una monitorización por medio de ping hacia el *Gateway* por un lapso de 3 horas, los resultados obtenidos se analizan en la sección de análisis de resultados. Para comprobar la conexión entre la red LoRa y el servidor en AWS se realizar de la misma forma una petición ICMP con ping hacia el DNS *wirelbits.ga* desde la consola del *Gateway LoRa*. El resultado de la Figura 112 determina que no existen paquetes perdidos y el tiempo de respuesta promedio es de 158 ms.

**Figura 112**

*Respuesta ICMP Gateway Servidor*

```
root@dragino-1f6f78:~# ping wirelbits.ga
PING wirelbits.ga (54.233.229.244): 56 data bytes
64 bytes from 54.233.229.244: seq=0 ttl=40 time=166.840 ms
64 bytes from 54.233.229.244: seq=1 ttl=40 time=147.022 ms
64 bytes from 54.233.229.244: seq=2 ttl=40 time=166.993 ms
64 bytes from 54.233.229.244: seq=3 ttl=40 time=148.853 ms
64 bytes from 54.233.229.244: seq=4 ttl=40 time=170.741 ms
64 bytes from 54.233.229.244: seq=5 ttl=40 time=149.632 ms
64 bytes from 54.233.229.244: seq=6 ttl=40 time=170.961 ms
64 bytes from 54.233.229.244: seq=7 ttl=40 time=149.955 ms
64 bytes from 54.233.229.244: seq=8 ttl=40 time=171.377 ms
64 bytes from 54.233.229.244: seq=9 ttl=40 time=147.162 ms
--- wirelbits.ga ping statistics ---
10 packets transmitted, 10 packets received, 0% packet loss
round-trip min/avg/max = 147.022/158.953/171.377 ms
```

Por medio del comando *tracert* se realiza la comprobación del tránsito realizado en la red y determinar el número de saltos que se producen hasta llegar al servidor. En la Figura 113, se observa que se realiza un total de 28 saltos, también se observa que existen redes que son de carácter privador y no visibles por la implementación de firewalls para seguridad informática.

Figura 113

Tracert dirigido al servidor "wirelbits.ga"

```
C:\Users\ADMIN>tracert wirelbits.ga

Trazo a la dirección wirelbits.ga [54.233.229.244]
sobre un máximo de 30 saltos:

  1  *          118 ms    2 ms  10.130.1.1
  2  3 ms      1 ms     1 ms  192.168.100.1
  3  11 ms     6 ms     6 ms  host-181-198-147-129.netlife.ec [181.198.147.129]
  4  10 ms     6 ms    16 ms  10.201.203.54
  5  12 ms     6 ms     8 ms  10.201.203.78
  6  4 ms      4 ms     4 ms  10.201.203.13
  7  6 ms      8 ms     5 ms  10.201.222.8
  8  4 ms      4 ms     4 ms  192.168.0.42
  9  5 ms      8 ms     5 ms  host-181-39-98-21.telconet.net [181.39.98.21]
 10  *          *          *      Tiempo de espera agotado para esta solicitud.
 11  21 ms    20 ms    26 ms  99.83.64.132
 12  24 ms    20 ms    24 ms  150.222.9.20
 13  50 ms    60 ms    29 ms  150.222.9.35
 14  *          *          *      Tiempo de espera agotado para esta solicitud.
 15  139 ms   142 ms   140 ms  52.93.132.51
 16  *          *          *      Tiempo de espera agotado para esta solicitud.
 17  161 ms   163 ms   161 ms  54.240.244.166
 18  *          *          *      Tiempo de espera agotado para esta solicitud.
 19  142 ms   141 ms   141 ms  150.222.69.114
 20  141 ms   141 ms   141 ms  54.240.244.207
 21  151 ms   146 ms   152 ms  54.240.244.100
 22  *          *          *      Tiempo de espera agotado para esta solicitud.
 23  *          *          *      Tiempo de espera agotado para esta solicitud.
 24  *          *          *      Tiempo de espera agotado para esta solicitud.
 25  *          *          *      Tiempo de espera agotado para esta solicitud.
 26  *          *          *      Tiempo de espera agotado para esta solicitud.
 27  *          *          *      Tiempo de espera agotado para esta solicitud.
 28  254 ms   148 ms   159 ms  ec2-54-233-229-244.sa-east-1.compute.amazonaws.com [54.233.229.244]

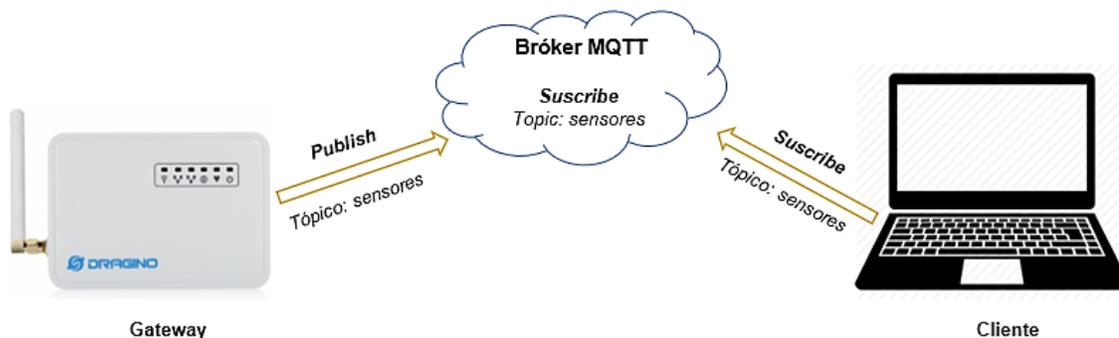
Trazo completa.
```

### Conectividad Gateway-Bróker-Plataforma IoT

La Figura 114 muestra el escenario de prueba para la verificación de comunicación entre el Gateway, el bróker y la plataforma desarrollada.

Figura 114

Escenario transmisión de los datos



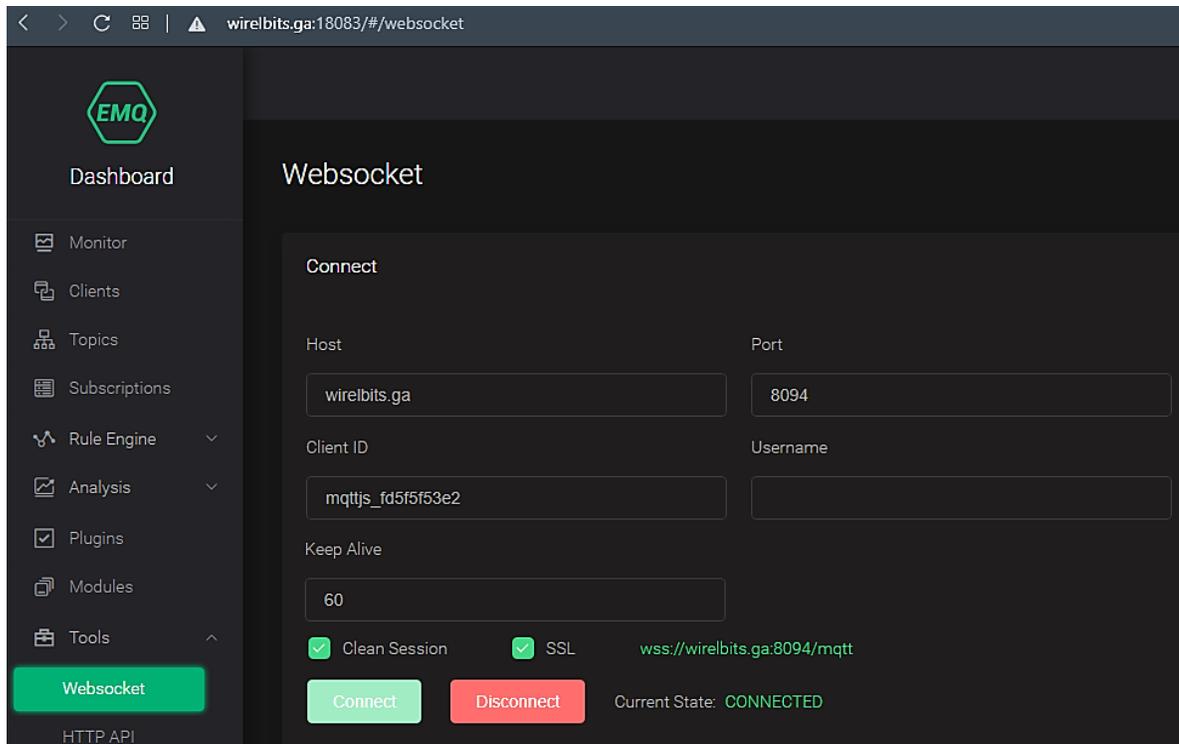
Al recibir los datos vía LoRa, el *Gateway* transmite los datos al bróker servidor según los parámetros configurados, por medio de los logs de la Figura 115 se observa la correcta comunicación entre ambos elementos.

### Figura 115

*LogRead de transmisión de datos al servidor*

```
user.notice root: [IoT.MQTT]:MQTT Publish Case: 4
user.notice root: [IoT.MQTT]:MQTT Publish Parameters
user.notice root: [IoT.MQTT]:server[-h]: wirelbits.ga
user.notice root: [IoT.MQTT]:port[-p]: 1883
user.notice root: [IoT.MQTT]:user[-u]:
user.notice root: [IoT.MQTT]:pass[-P]:
user.notice root: [IoT.MQTT]:pub_qos[-q]: 0
user.notice root: [IoT.MQTT]:cafile[--cafile]:
user.notice root: [IoT.MQTT]:cert[--cert]: /etc/iot/cert/
user.notice root: [IoT.MQTT]:key[--key]: /etc/iot/cert/
user.notice root: [IoT.MQTT]:clientID[-i]: dragino_gateway
user.notice root: [IoT.MQTT]:remote_id: 4567
user.notice root: [IoT.MQTT]:pub_topic[-t]: sensores
user.notice root: [IoT.MQTT]:decoder: Not Set
user.notice root: [IoT.MQTT]:mqtt_data[-m]: 18.0,85.0,6.33, 73.0,9.00,-88
```

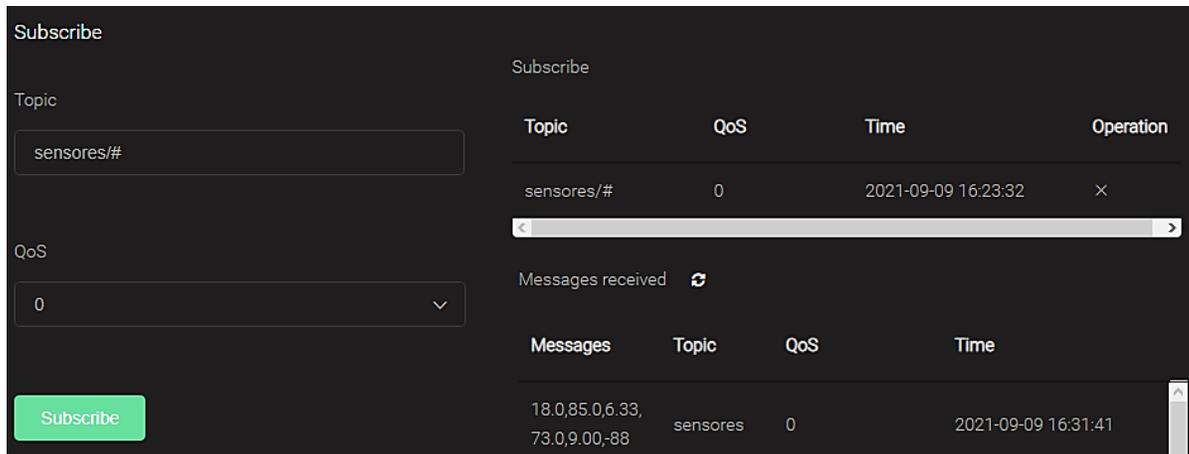
Por medio de la herramienta Websocket del bróker se verifica la recepción de mensajes, los parámetros de configuración para esta prueba se observan en la Figura 116.

**Figura 116***Configuración de Websocket*

En la sección de suscribe, se establece el tópic con el cual transmite los datos el *Gateway* y la calidad de servicio, luego de realizar la suscripción en la Figura 117 se observa la recepción de los mensajes en el bróker.

**Figura 117**

*Llegadas de mensajes al bróker*



Se accede a la plataforma IoT por medio de la dirección <https://wirelbits.ga>. Se accede al dashboard y desde la consola se visualiza la llegada de mensajes y la presentación en tiempo real en la página web como se observa a continuación:

**Figura 118**

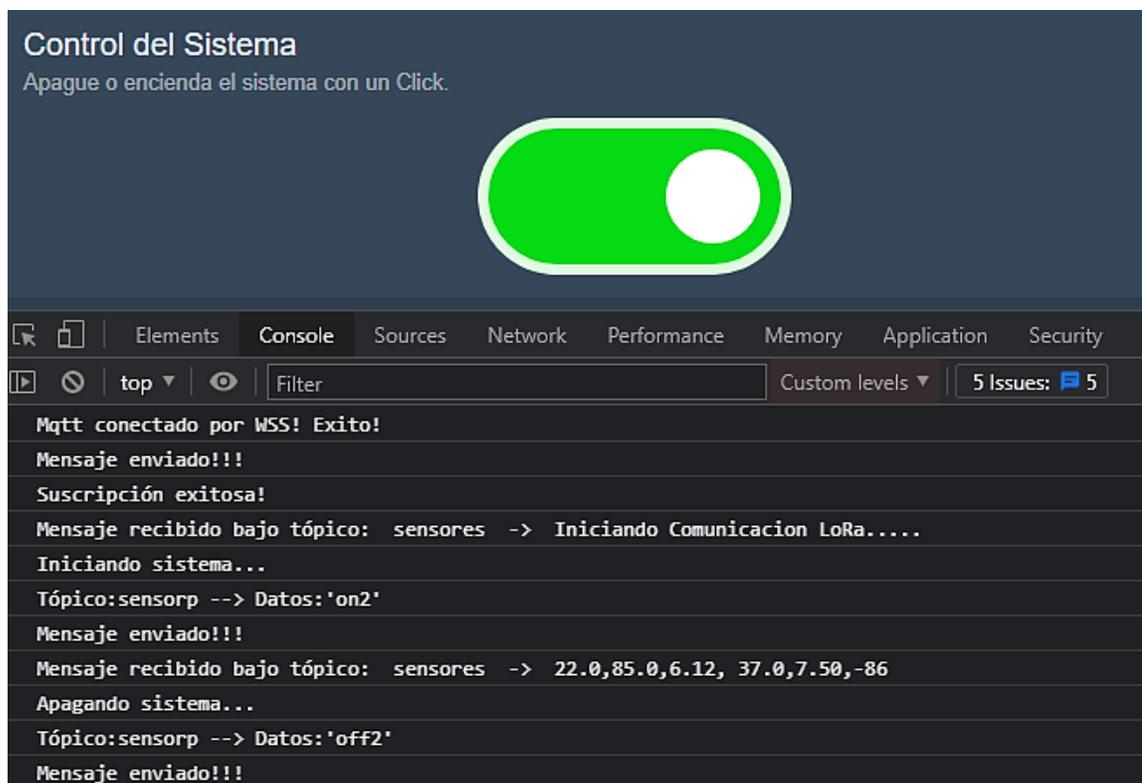
*Visualización de los datos en tiempo real*



Para iniciar o detener el sistema se dispone de un botón, por medio de la consola del navegador es posible verificar el envío de los mensajes como se visualiza en la Figura 119.

**Figura 119**

*Envío de mensajes desde la plataforma*



Quando se recibe un mensaje, en los logs del *Gateway* se aprecia la trama metadata en la cual se define la modulación utilizada, los parámetros de la capa física configurada y los datos transmitidos como se aprecia en la Figura 120.

Figura 120

## LogRead del Gateway LoRa

```

FreqINFO:
RF_RADIO struct: spiport=4, freq=915000000, sf=7
LoRa Gateway service Mode=mqtt, gatewayID=a840411f6f784150, server=nam1.cloud.thethings.network

Logread RxTxJson:
Receive(HEX):3c343536373e31382e302c38352e302c362e33332c2036372e302c392e32352c2d3934

(TXPKT): [push] [{"txpk":
{"imme":true,"tmst":204492134,"freq":915.000000,"rfch":0,"pove":20,"modu":"LORA","datr":"SF7BW125","codr":"4/5","ipol":false,"size":4,"ncrc":true,"data":"off2"}]}

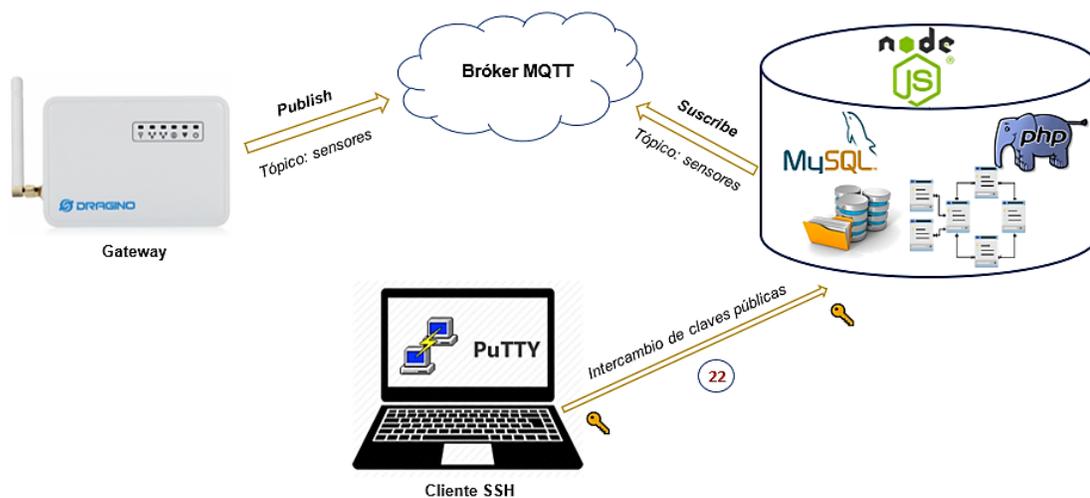
(TXPKT): [push] [{"txpk":
{"imme":true,"tmst":205881718,"freq":915.000000,"rfch":0,"pove":20,"modu":"LORA","datr":"SF7BW125","codr":"4/5","ipol":false,"size":3,"ncrc":true,"data":"on2"}]}

```

## Almacenamiento en la BD

Figura 121

## Escenario de verificación de almacenamiento



La prueba de almacenamiento tiene el propósito de comprobar que exista un óptimo almacenamiento de los datos en el servidor de AWS, para ello por medio de

SSH se realiza la conexión remota a la instancia en la cual se encuentra alojada la BD y desde consola se observa los logs de Node.JS como se observa a continuación:

**Figura 122**

*Logs de Node.JS*

```
root@wirelbits:/home/admin/web/wirelbits.ga/public_html/node# pm2 log
/root/.pm2/logs/index-out.log last 15 lines:
0|index | Datos Enviados a THINKSPEAK WIRELBITS...!!!
0|index | Datos Enviados a THINKSPEAK
0|index | Mensaje recibido desde -> sensores Mensaje -> 21.0,85.0,6.13, 40.0,7.75,-86
0|index | Datos Enviados a THINKSPEAK WIRELBITS...!!!
0|index | Datos Enviados a THINKSPEAK
0|index | Encendiendo bomba...!!!
0|index | Fila insertada en la BD correctamente
0|index | Mensaje recibido desde -> sensores Mensaje -> 21.0,85.0,6.15, 40.0,7.75,-86
```

Por medio del software Heidi SQL se conecta a la base de datos y se observa la correcta inserción de la información, así como también la fecha y hora.

**Figura 123**

*Inserción de los datos en la BD*

Host: wirelbits.ga Base de datos: admin\_lorawan Tabla: tablaThingSpeak Datos Consulta

admin\_lorawan.tablaThingSpeak: 6.440 filas en total (aproximadamente) >> Siguintes

id	ph	humedad	temperatura	distancia	snr	rssi	fecha_hora
6.416	6,19	40	21	85	7,75	-86	2021-09-13 16:43:24
6.417	6,14	40	21	85	7,75	-86	2021-09-13 16:43:41
6.418	6,17	39	21	85	7,75	-86	2021-09-13 16:44:00
6.419	6,13	40	21	85	7,75	-86	2021-09-13 16:44:23
6.420	6,15	40	21	85	7,75	-86	2021-09-13 16:44:40

### Visualización de los datos estadísticos, históricos y SetPoints

Esta prueba tiene el propósito de verificar el correcto funcionamiento de la plataforma IoT desarrollada, para ello se ingresa en la sección estadísticas y se visualiza la información transmitida en tiempo real por los sensores actualizada cada 5 segundos, cabe mencionar que las gráficas pueden ser descargadas en los formatos:

*png, jpeg, pdf y svg.*

Figura 124

Estadísticas en tiempo real de los datos



La plataforma también dispone de gadgets que permiten visualizar de una manera más práctica los datos transmitidos por los sensores.

### Figura 125

#### *Gadgets de visualización*



Para la consulta de datos se ingresa a la sección de históricos de la variable física que se desea analizar. Como ejemplo se ingresa a la sección de históricos de temperatura y se observa la siguiente interfaz gráfica.

**Figura 126***Interfaz gráfica de históricos*

Bienvenido a WireBoard IoT  
Tecnología IoT con Lora

Gestionando: Hidroponia

## Historicos Temperatura

**Precaución**  
Antes de Buscar actualice los datos

Actualizar Datos

**Búsqueda**  
Seleccione la fecha y hora que desea consultar:

FECHA INICIO: 03/07/2021

FECHA FIN: 31/07/2021

HORA INICIO: 00:00

HORA FIN: 00:00

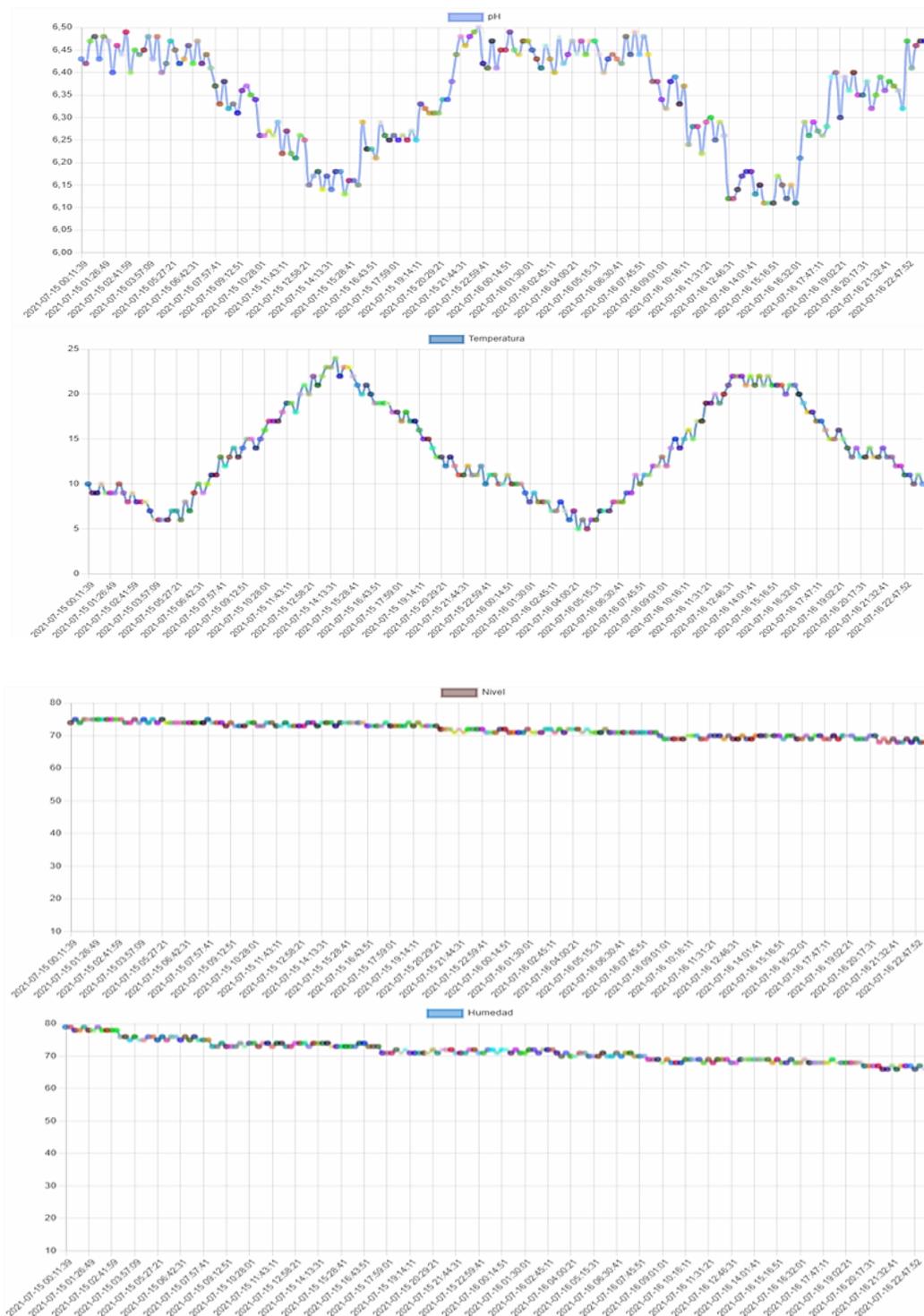
**Gráfica**  
Seleccione el tipo de gráfico que desea realizar:

Grafico Barras   Grafico Horizontal   Grafico Líneas   Grafico PIE   Grafico Scatter

La interfaz permite definir el rango de búsqueda (fechas y horas) deseado, así como también el tipo de gráfica a generar. Como prueba de funcionamiento se realiza la búsqueda del día 15 al 16 de julio y los resultados obtenidos se presentan a continuación:

Figura 127

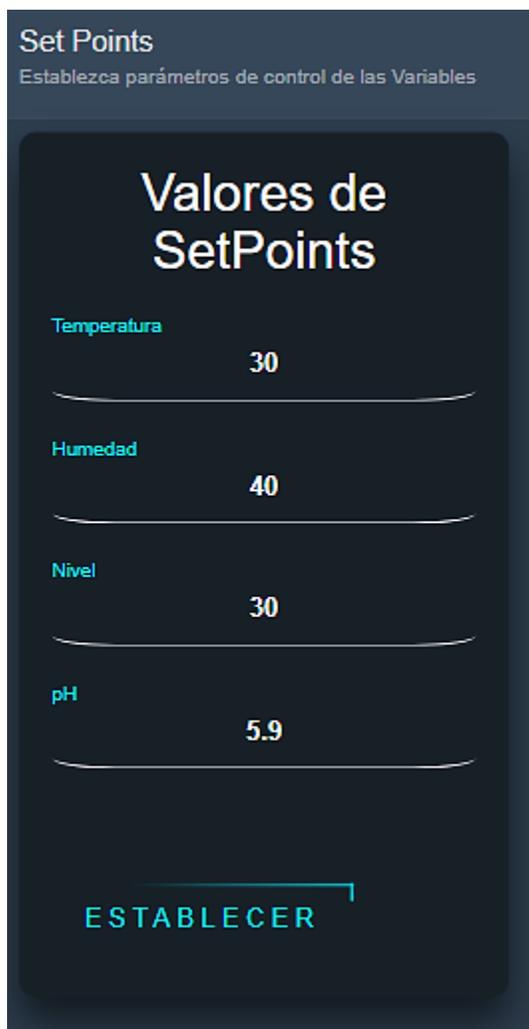
Datos históricos de las variables físicas



Para la inserción de *SetPoints* que permiten la emisión de alertas en tiempo real, desde el dashboard se realiza la colocación de los valores máximos y mínimos de comparación por medio del siguiente formulario:

### Figura 128

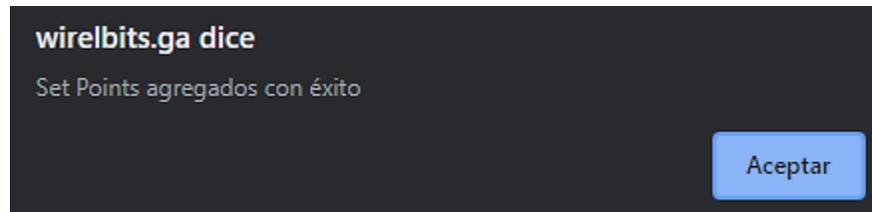
Formulario de *SetPoints*



The image shows a mobile application interface for setting control parameters. At the top, it says 'Set Points' and 'Establezca parámetros de control de las Variables'. Below this is a dark blue rounded rectangle containing the title 'Valores de SetPoints'. There are four rows, each with a variable name on the left and a numerical value in the center, separated by a horizontal line. The variables and their values are: Temperatura (30), Humedad (40), Nivel (30), and pH (5.9). At the bottom of the rectangle is a large, teal-colored button with the text 'ESTABLECER'.

Variable	Valor
Temperatura	30
Humedad	40
Nivel	30
pH	5.9

Una vez definido los puntos de referencia, por medio del botón establecer se realiza el proceso de inserción en la base de datos como se observa en la Figura 129.

**Figura 129***Inserción de los SetPoints*

id	temperatura	humedad	nivel	ph	fecha_hora
2	19	75	80	6,8	2021-06-07 23:49:18
3	30	40	50	6,59	2021-09-14 12:41:58
4	30	40	30	5,9	2021-09-14 15:01:45

**Alertas en tiempo real**

En la aplicación de *Node.JS* se realiza una comparativa por medio de consultas SQL, entre los últimos valores de SetPoints y datos de las variables monitorizadas que han sido insertados en la BD. Con el uso de la plataforma Twilio se envía mensajes SMS, como ejemplo se envía un SMS cuando el pH recibido por los sensores es mayor al umbral. En la Figura 130 se observan los logs de *Node.Js* con el proceso de envío en tiempo real de las alertas y los SMS recibidos en el teléfono del cliente configurado.

**Figura 130***Logs de las alertas SMS enviadas*

```

root@wirelbits:/home/admin/web/wirelbits.ga/public_html/node# node appS.js
Conexión MQTT Exitosa!
Subscripción exitosa!
Mensaje recibido desde -> sensores Mensaje -> 20.0,85.0,6.24, 55.0,8.00,-90
SetPoints:..30 40 30 5.9
Temperatura Correcta
Humedad Correcta
Nivel de Agua Correcto
ALERTA SMS ==>> ENVIADA

```



### **Análisis de Resultados**

Se requiere mencionar que se dispone únicamente de un nodo final, la adquisición de datos se realiza por un periodo de 24 horas durante 4 semanas, la transmisión y almacenamiento de los datos se realiza cada 15 minutos en base al algoritmo de la Figura 63 y la explicación del criterio de diseño.

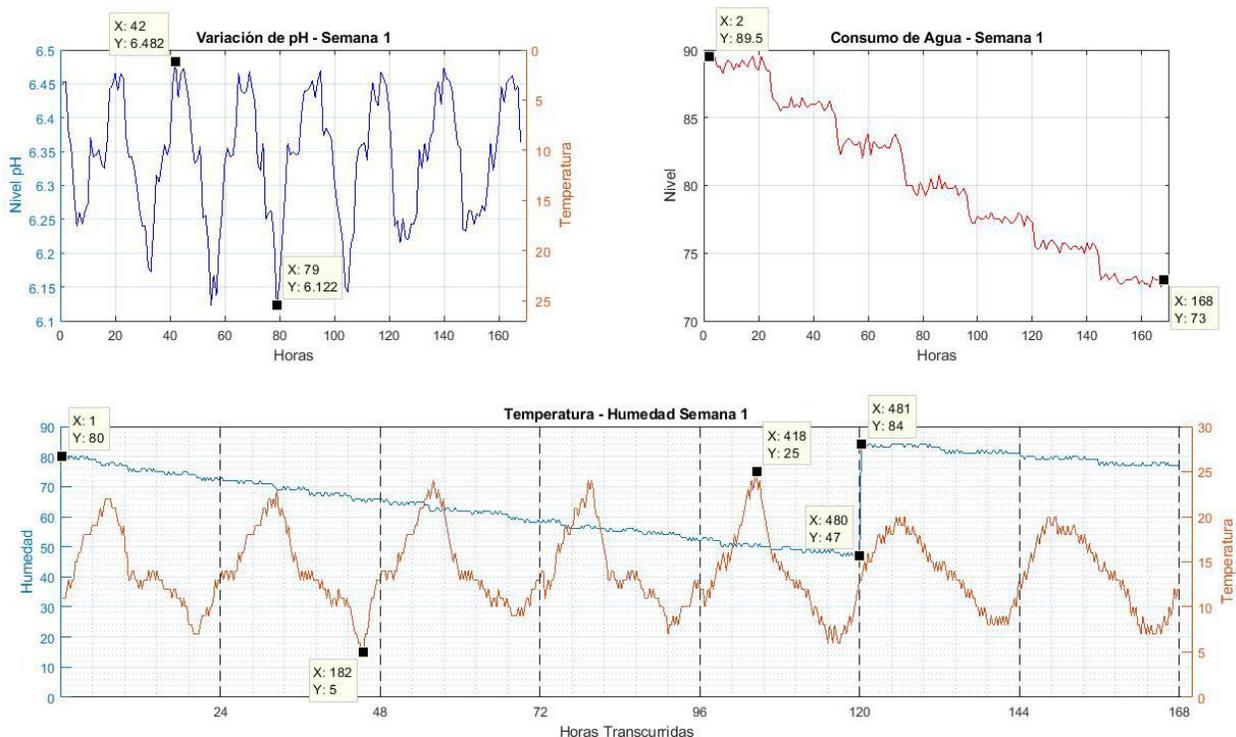
### **Variables Físicas**

En la Figura 131, se observa las señales físicas obtenidas del proceso de monitorización de la semana 1, los niveles de pH oscilan entre 6.48 y 6.12 conforme varía la temperatura ambiental. La gráfica del consumo de agua refleja una disminución de un 16.5% en el nivel de solución nutritiva del contenedor lo que representa un consumo de 8.5 litros de agua en la semana, la temperatura en el interior del invernadero registra variaciones entre 25 y 5 grados mientras que el porcentaje humedad alcanza un valor de 47 transcurrido los 5 primeros días, tiempo en cual se realiza el riego de agua en los semilleros debido a que en base a la experimentación el

porcentaje de humedad no debe ser inferior al 30% para una correcta germinación de las plantas.

**Figura 131**

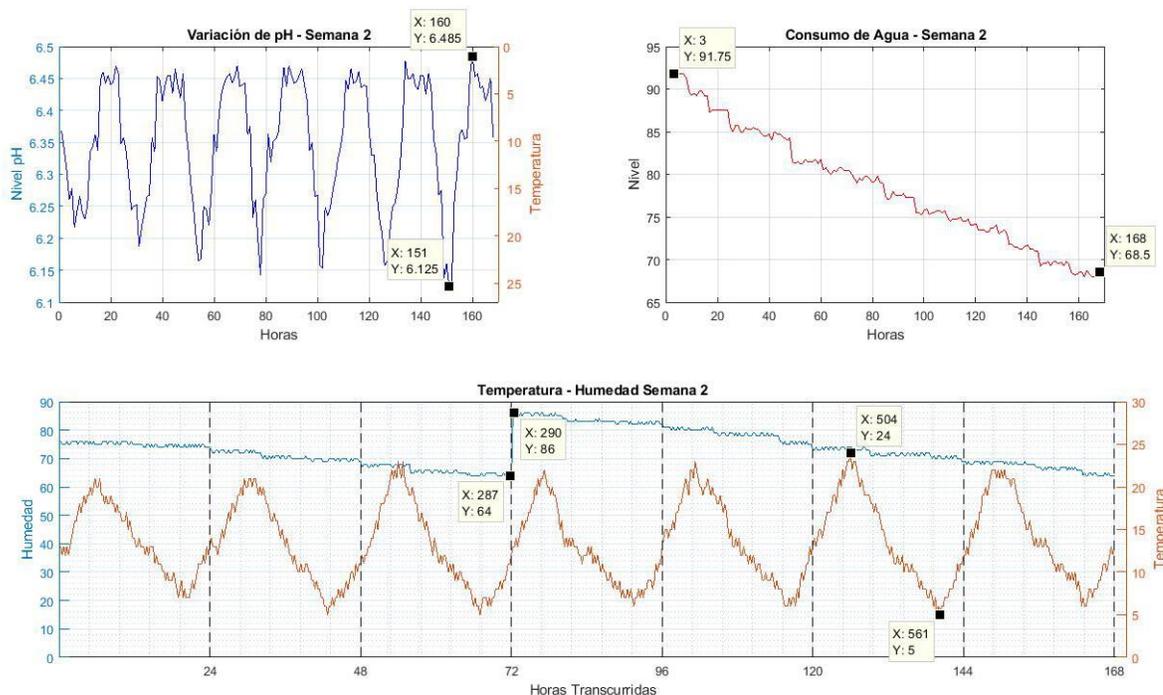
*Señales físicas obtenidas semana 1*



La Figura 132 representa las señales físicas obtenidas del proceso de monitorización de la semana 2, los niveles de pH presentan variaciones similares entre 6.48 y 6.12. La gráfica del consumo de agua refleja una disminución de un 23.25% en el nivel de solución nutritiva del contenedor lo que representa un consumo de 10.75 litros de agua, la temperatura en el interior del invernadero registra variaciones entre 24 y 5 grados mientras que el porcentaje humedad alcanza un valor de 64 transcurridas 72 horas de la segunda semana. Pasado los 3 días se realiza el riego de agua en los semilleros por lo que el sensor registra un porcentaje de 86% de humedad.

Figura 132

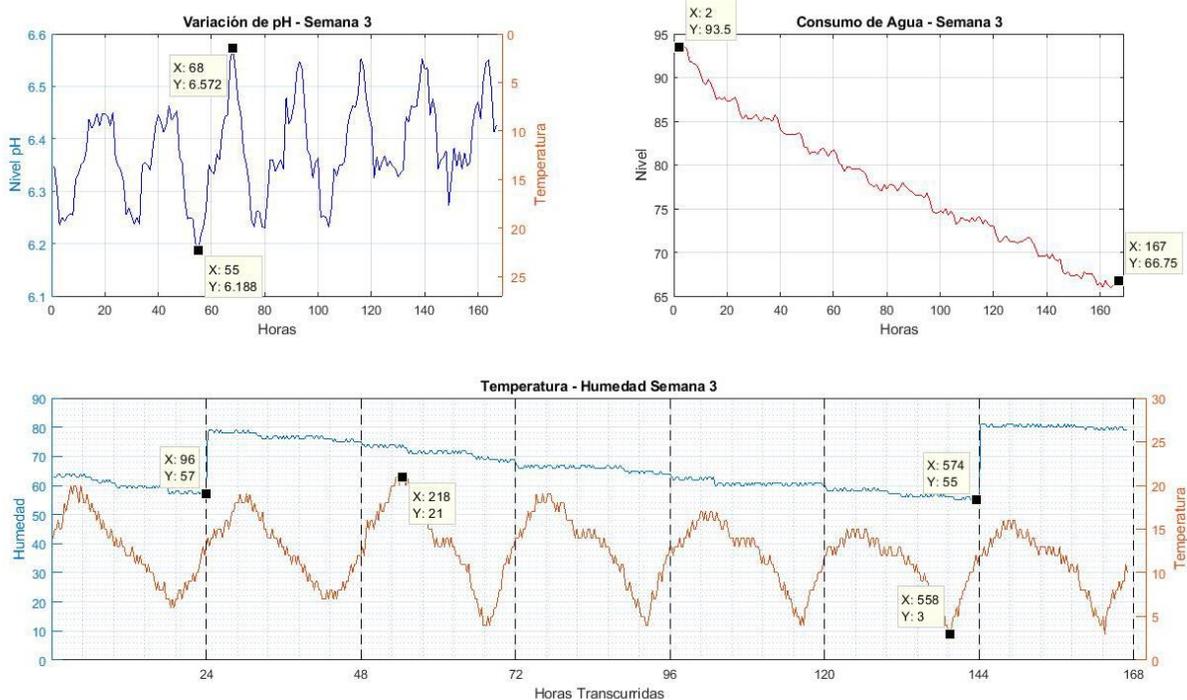
Señales físicas obtenidas semana 2



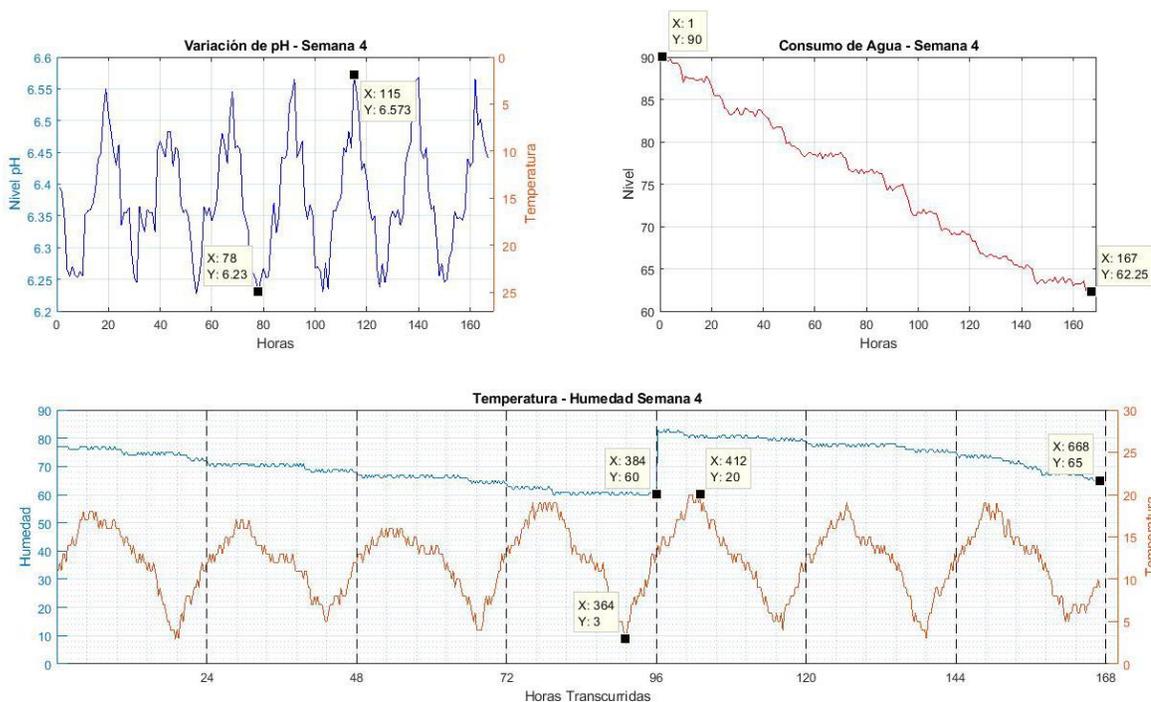
En la Figura 133, se observa las señales físicas del proceso de monitorización transcurridas 3 semanas, se registra una variación entre 6.57 y 6.18, el aumento del nivel de pH se debe al registro de menores temperaturas. La gráfica del consumo de agua refleja una disminución de un 26.75% en el nivel de solución nutritiva del contenedor lo que representa un consumo de 11.6 litros de agua. La temperatura en el interior del invernadero registra variaciones entre 21 y 3 grados mientras que el porcentaje humedad alcanza un valor de 57 transcurridas 24 horas, sin embargo, al haber transcurrido 5 días desde el último riego en los semilleros, se humedece la tierra para propiciar un ambiente adecuado de germinación.

**Figura 133**

*Señales físicas obtenidas semana 3*



Luego de 4 semanas de monitorización, en la Figura 134 se observa que los niveles de pH presentan variaciones comprendidas entre 6.57 y 6.23. La gráfica del consumo de agua refleja una disminución de un 27.75% en el nivel de solución nutritiva del contenedor lo que indica un consumo de 13.9 litros de agua. Con respecto a la temperatura, en el interior del invernadero se presenta variaciones entre 20 y 3 grados y con respecto al porcentaje de humedad se obtiene valores de 60% y 64% transcurridas 96 y 168 horas respectivamente.

**Figura 134***Señales físicas obtenidas semana 4***Semana 1**

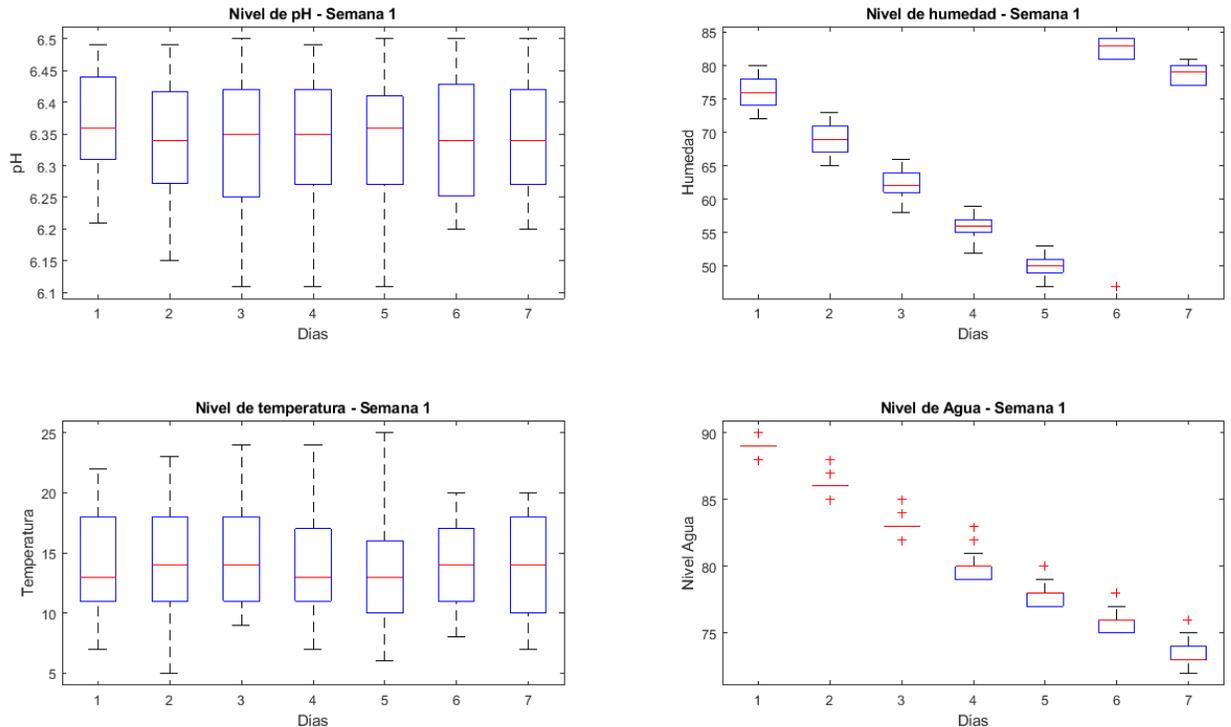
En la Figura 135, se observa la evolución del crecimiento en la primera semana de realizado el trasplante del semillero al sistema hidropónico. Es importante verificar que el agua recorra de manera lenta por los tubos para garantizar la correcta absorción de nutrientes, lo cual se evidencia en el aumento de las hojas de lechuga y tomate caracterizadas por un color amarillo claro. En esta semana el monitoreo de los niveles de pH debe estar comprendido entre 6.1 y 6.5, cabe mencionar que el color amarillento de las hojas se debe al estrés producido por el cambio del suelo al sistema hidropónico.

### Figura 135

*Proceso de crecimiento de las plantas*



En la Figura 136, por medio de los diagramas de caja se puede apreciar que los niveles de pH a lo largo de la semana 1 varían entre 6.11 y 6.5, los días con un pH más ajustado fueron el primer y quinto día con variaciones entre 6.33 y 6.44. El nivel de humedad presenta menos variaciones a lo largo de los 7 días, el primer día hay un consumo mayor de agua y evaporación por lo que la humedad de la tierra varía de 80 a 73, al cabo del quinto día se alcanza un nivel de tierra semi húmeda con un nivel de variación de 54 a 51 por lo que en el sexto día se agrega nuevamente agua. La temperatura presenta rangos más amplios de variación, el día más caluroso es el quinto día con temperaturas variables entre 25 y 3 grados centígrados. El nivel del agua es mucho más ajustado y se aprecia como a lo largo de la semana se incrementa, el séptimo día es el día de mayor consumo de agua por parte de las plantas con un aproximado de 1.8 litros.

**Figura 136***BoxPlot Variables Físicas - Semana 1***Semana 2**

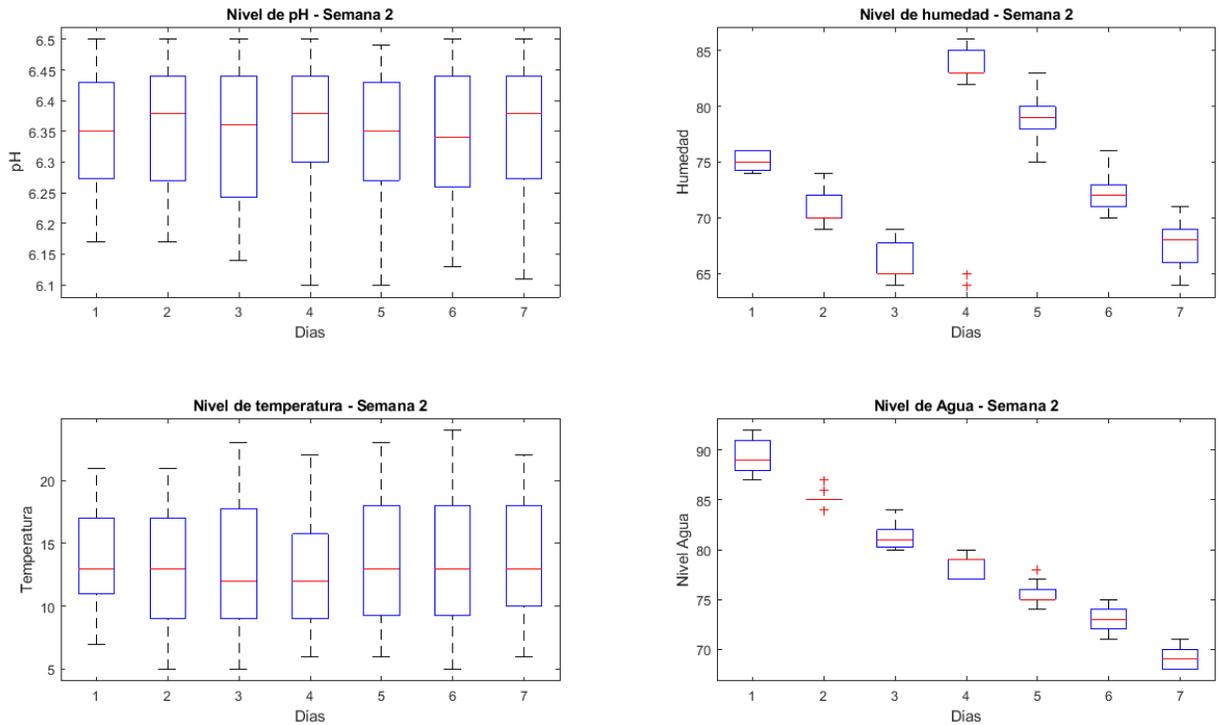
Los cambios más evidentes en la segunda semana de trasplante es la coloración más verdosa en las hojas de las plantas debido a la mejor asimilación de los nutrientes, se aprecia un crecimiento más vigoroso y aumento de hojas. En esta semana es importante verificar que en la solución nutritiva no exista la presencia de algas al interior del invernadero también se observa el surgimiento de trips (insectos pequeños) en algunas hojas, razón por la cual se realiza la fumigación con extracto de ortiga para disminuir su población debido a que estos insectos causan perforaciones y cuando la infestación es demasiado grave provoca que las hojas se arruguen y marchiten.

**Figura 137**

*Proceso de crecimiento de las plantas*



En la figura 138 se aprecia que el nivel de pH es más ajustado en el segundo día con variaciones entre 6.35 y 6.44. El nivel de humedad presenta un aumento de variaciones a lo largo de los 7 días, en el séptimo día hay un consumo mayor de agua en los semilleros evidenciado en una humedad de la tierra que varía entre 74 a 64. De igual manera la temperatura presenta rangos más amplios de variación, el día más caluroso es el sexto día con temperaturas variables entre 24 y 4 grados centígrados. El nivel del agua es mucho más ajustado se aprecia como a lo largo de la semana se incrementa, el día de menor consumo es el noveno donde casi el nivel permanece en un 85% es decir en 37.8 litros lo que indica una absorción 2.5 litros de solución nutritiva.

**Figura 138***BoxPlot Variables Físicas - Semana 2***Semana 3**

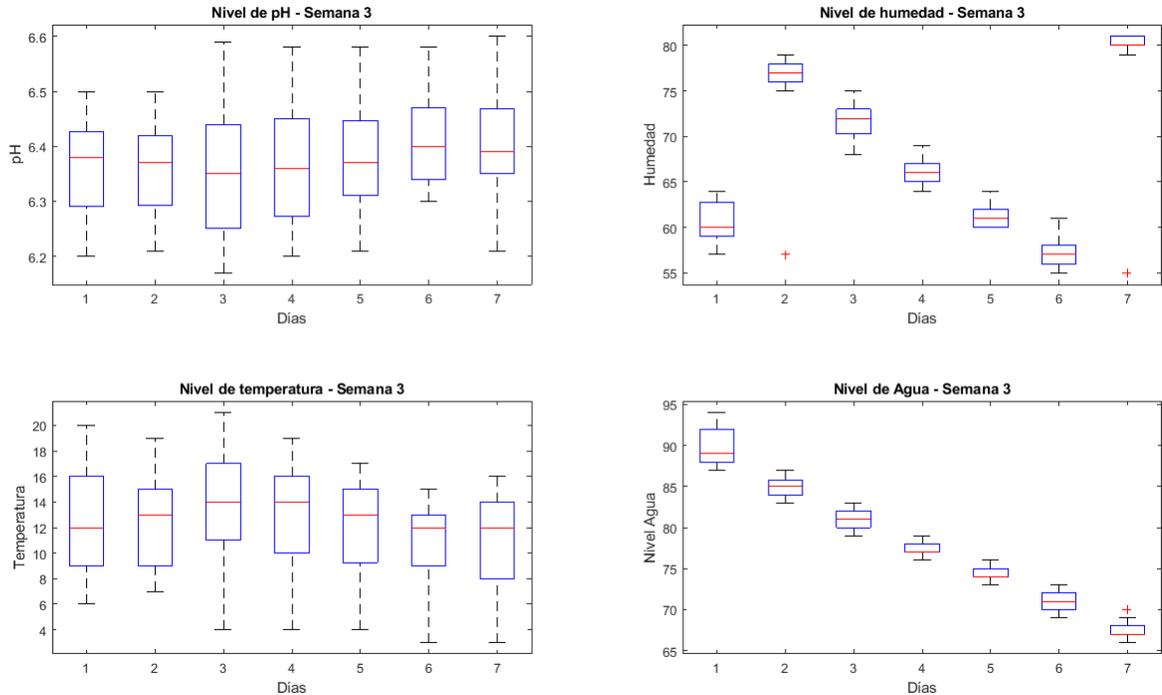
En la tercera semana se evidencia la proliferación de algas en el contenedor de la solución nutritiva por lo que se procede a cambiar aproximadamente 20 litros de agua. El notable crecimiento de las hojas se debe al control de pH, por medio de la monitorización se detecta un aumento de los niveles de pH, para regular esta variable se coloca una malla negra y se agrega zumo de limón puro. Otro cambio notorio es el aumento y crecimiento de las raíces, lo que hay que tomar en cuenta para evitar obstrucciones en la circulación de solución nutritiva en el sistema hidropónico. Las algas son otro inconveniente por lo que para reducir su proliferación se agrega mínimas cantidades de agua oxigenada.

**Figura 139**

*Proceso de crecimiento de las plantas*



Los diagramas de caja permiten determinar que los niveles de pH varían entre 6.17 y 6.6, el día con el pH más ajustado es el décimo sexto día con variaciones entre 6.2 y 6.5. El nivel de humedad presenta menos variaciones al cabo de 15 días se alcanza un nivel de 57% y hay mayor consumo de agua por parte de las plantas en los semilleros. La temperatura presenta rangos mucho más amplios sin embargo se evidencia días más fríos al registrar valores comprendidos entre 21° a 3° centígrados. En la tercera semana el consumo es mucho mayor pues se registra un consumo diario de aproximadamente 2 litros por día. Los días 16 y 21 se agrega agua a los semilleros debido al rápido decremento de los niveles de humedad al detectar niveles de 55% y 56, el nivel de agua es progresivo en 15 días se agrega 4 litros extra de solución alcanzado un total de 47 litros, la variación de los niveles de pH presenta oscilaciones entre 6.1 y 6.6.

**Figura 140***BoxPlot Variables Físicas - Semana 3***Semana 4**

En esta semana se caracteriza por un aumento importante en el grosor y número de raíces que obstruyeron la normal circulación de solución nutritiva, las lechugas alcanzan una altura de 50 cm mientras que los tomates de 60 cm, se nota que algunas hojas jóvenes se encuentran dobladas y arrugadas lo que evidencia una notable deficiencia de calcio razón por la cual se agregan 800 gramos extras de este componente a la solución nutritiva.

**Figura 141**

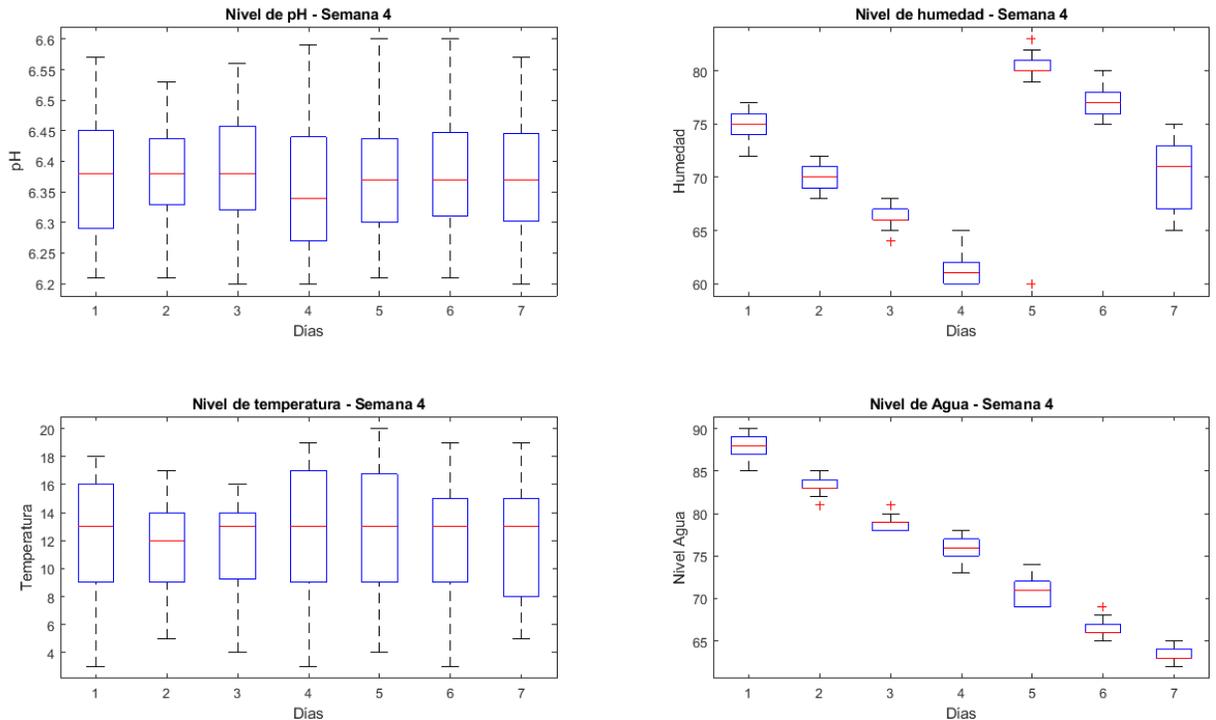
*Proceso de crecimiento de las plantas*



Los diagramas de caja establecen que los niveles de pH varían entre 6.1 y 6.6, el día con el pH más ajustado es el segundo día (vigésimo tercero) con variaciones entre 6.2 y 6.53, con respecto a los niveles de humedad hay un abrupto consumo en el último día de monitorización reduciéndose de 75% a 60%, existen algunos valores atípicos que se descartan para el análisis, de igual manera se verifica que la sensación térmica reduce pues se tiene rangos de temperatura de 20 a 3 grados centígrados. El consumo de nivel de agua es otro factor importante pues se evidencia que el consumo aproximado en esta semana es de 2.3 litros diarios.

**Figura 142**

*BoxPlot Variables Físicas - Semana 4*



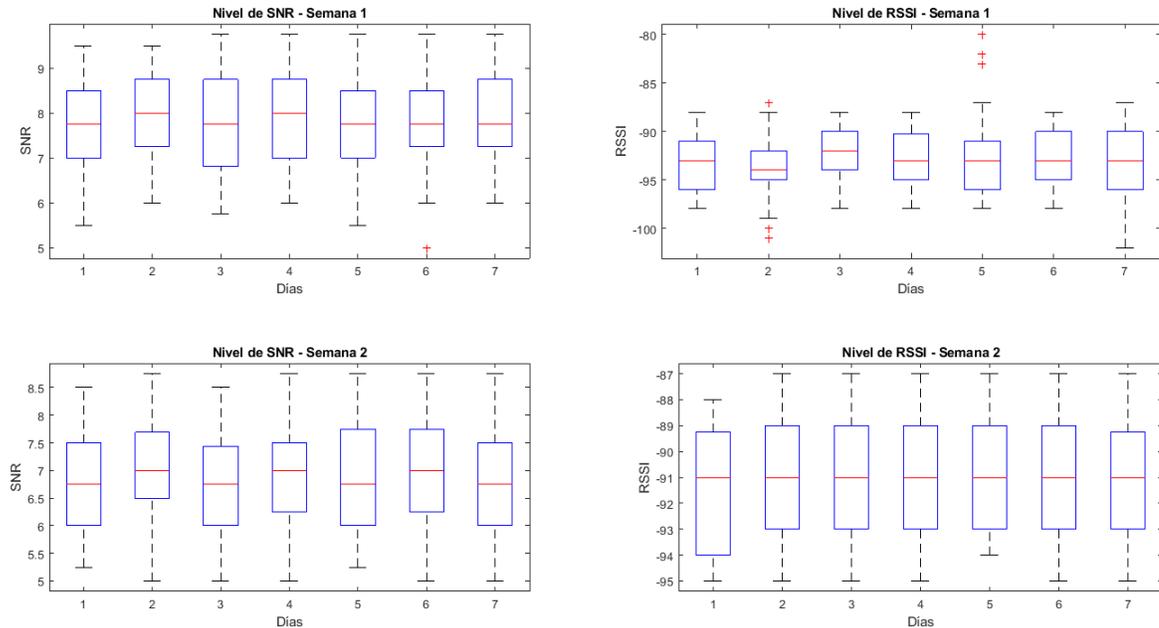
## Parámetros

En la Figura 143, se definen los indicadores de fuerza de señal recibida (*RSSI*, del inglés *Received Signal Strength Indicator*), que permite establecer el nivel de calidad con la que el receptor capta la señal. LoRa presenta valores típicos de *RSSI* que varían entre -120 dBm hasta -30 dBm, cabe mencionar que un valor cercano a cero indica que se tiene un nivel de señal muy fuerte. Otro factor importante es la relación señal a ruido (*SNR*, del inglés *signal to noise ratio*), definida como la proporción que existe entre las potencias de señal transmitida y de ruido. Si el *SNR* es mayor a cero, se establece que la señal recibida trabaja por encima del piso de ruido, LoRa presenta valores típicos de *SNR* que oscilan entre los -20 dB hasta 10 dB, valores cercanos a 10 dB indican que la

señal que se recibe presenta menos daños. Por medio de LoRa es posible realizar la demodulación de señales que operan por debajo del piso del ruido con valores comprendidos entre los -7.5 dB hasta los -20 dB. El diagrama de caja indica que los niveles de *SNR* obtenidos en la semana 1 y 2 se encuentra comprendidos entre 9.75 y un mínimo de 5 con una media de 7.82 y una mediana de 7.75 al ser la mediana mayor a la media los valores atípicos se encuentra hacia abajo del diagrama de caja, mientras que el RSSI presenta un valor máximo de -80 y mínimo de -103 con una media de -93.02 y mediana de -93 al ser la mediana casi idéntica a la moda los datos son simétricos, existen valores atípicos en el diagrama que son descartados del análisis. En la semana 2 los valores obtenidos son mucho más estables, el máximo valor de *SNR* recibido es de 8.75 mientras que el mínimo de 5, la media calculada es de 6.88 y la mediana de 7, al ser la media mayor a la mediana los valores atípicos se encuentran arriba del diagrama de caja, los niveles de RSSI presenta variaciones entre -87 a -95 con una media de -91.12 y una mediana de -91 lo que determina casi una simetría.

**Figura 143**

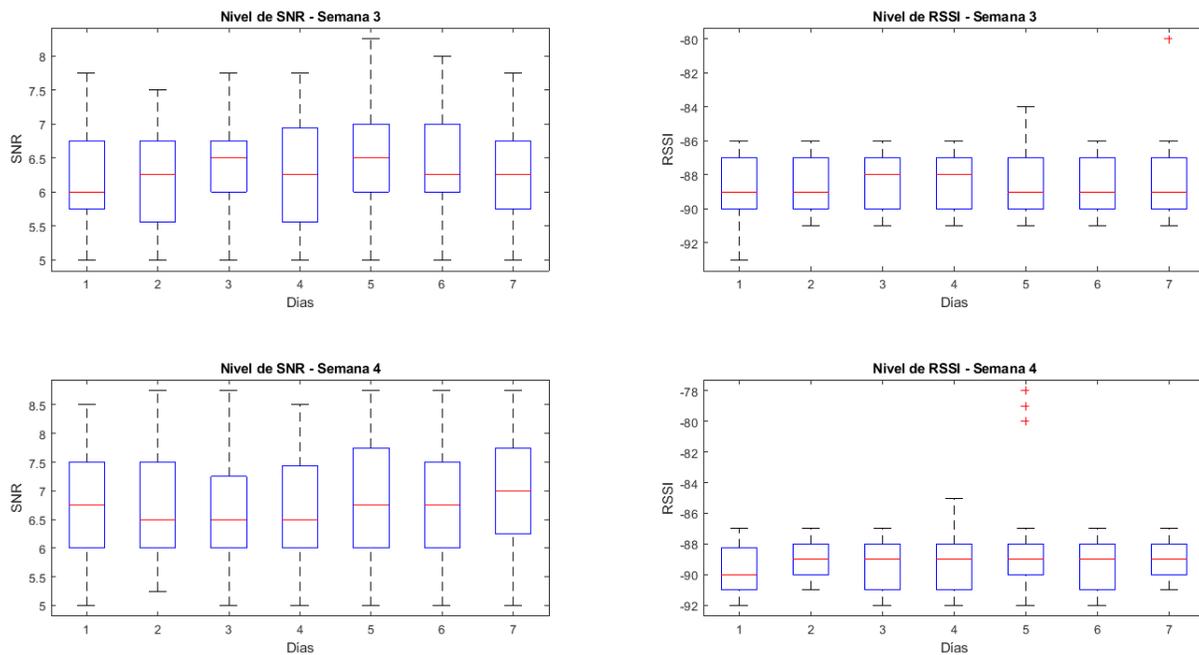
*SNR y RSSSI obtenidos - Semana 1 y 2*



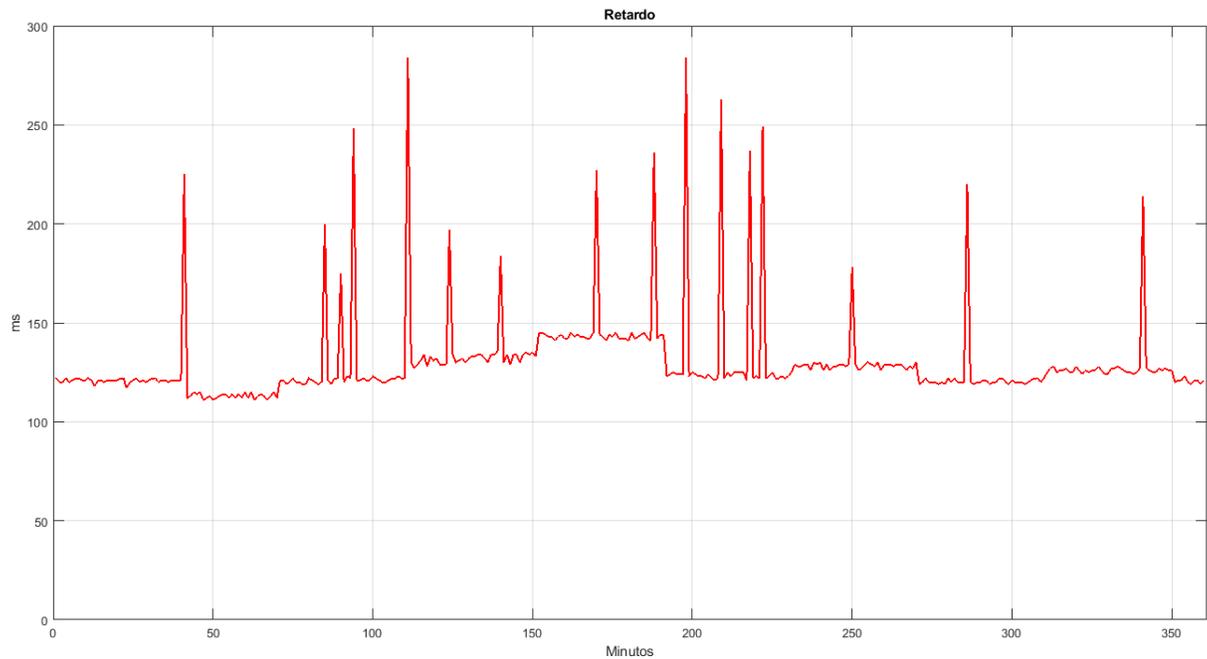
El diagrama de caja de la Figura 144, indica que el nivel de SNR presenta variaciones entre 8.25 y 5 su media se encuentra ubicada entre 6.31 y su mediana de 6.25 al ser la media mayor a la mediana los valores atípicos se encuentran hacia arriba, el nivel de RSSI varía entre -80 a -93 con una media de -88.5 y una mediana de -89-, el valor atípico varía de manera significativa a la media es por ello que es importante determinar la mediana para el análisis. En la semana 4 los niveles de SNR se encuentran entre 8.75 y 5, sin embargo, la media y la mediana tiene un valor de 6.75 lo que indica que existe una simetría, los valores de RSSI oscilan entre -78 y -92 y claramente se evidencia una mayor estabilidad en los parámetros de enlace pues en esta semana tanto la media como la mediana tienen un valor de -89.

**Figura 144**

SNR y RSSSI obtenidos - Semana 3 y 4



En la Figura 145, se pueden apreciar los retardos de la red inalámbrica obtenidos por medio de un ping sostenido, se establece que el máximo retardo alcanzado es de 284 ms y el mínimo de 111 ms, el valor promedio es de 129.95 ms mientras que la mediana de 124 ms, la media se ve claramente afectada por los picos de retardo alcanzados. Los tiempos de retardo obtenidos garantizan óptimas transmisiones en tiempo real, pues en base a la investigación realizada por (Liang, Zhao, & Wang, 2020) con una potencia de transmisión de 20 dBm se registran valores de retardo de hasta 600 ms con una media de 584 ms en interiores.

**Figura 145***Retardos*

En base a la Figura 146 es posible definir el número de paquetes perdidos, se evidencia que el primer mensaje se recibe el 3 de julio de 2021 a las 08:04, mientras que el último mensaje se recibe el 30 de julio a las 22:47 lo que deja un total de 2.608 paquetes de datos recibidos. En la Figura 147, se aprecia que el porcentaje de paquetes perdidos es cercano al 3% y el total de paquetes recibidos es del 97% lo que garantiza que el sistema es apropiado para transmisiones en tiempo real al tener pérdida de datos inferiores al 5%.

**Figura 146**

*Paquetes perdidos*

Host: 15.228.145.190 Base de datos: admin\_lorawan Tabla: tabla\_sensor Datos Consulta

admin\_lorawan.tabla\_sensor: 3.550 filas en total (aproximadamente), limitado a 1.000

id	ph	humedad	temperatura	distancia	snr	rsi	fecha_hora
1	6,44	80	11	90	8	-93	2021-07-03 08:04:17
2	6,42	80	11	89	9,5	-90	2021-07-03 08:19:19

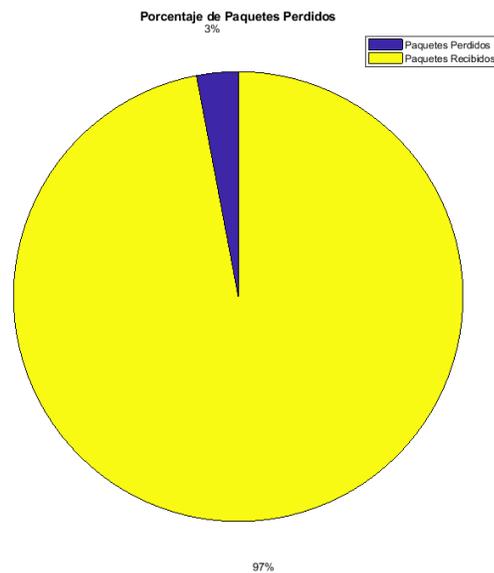
Host: 15.228.145.190 Base de datos: admin\_lorawan Tabla: tabla\_sensor Datos Consulta

admin\_lorawan.tabla\_sensor: 3.844 filas en total (aproximadamente)

id	ph	humedad	temperatura	distancia	snr	rsi	fecha_hora
2.596	6,53	66	6	62	7,5	-112	2021-07-30 19:47:16
2.597	6,41	66	7	62	8,25	-89	2021-07-30 20:02:18
2.598	6,41	65	7	62	7,25	-113	2021-07-30 20:17:20
2.599	6,4	66	8	63	6,5	-92	2021-07-30 20:32:22
2.600	6,5	66	8	63	9,5	-92	2021-07-30 20:47:24
2.601	6,5	65	9	62	7,25	-113	2021-07-30 21:02:26
2.602	6,41	65	9	62	9,5	-100	2021-07-30 21:17:28
2.603	6,45	65	9	63	6,5	-89	2021-07-30 21:32:30
2.604	6,47	65	10	62	8,75	-106	2021-07-30 21:47:32
2.605	6,43	65	9	62	8,75	-93	2021-07-30 22:02:34
2.606	6,48	65	10	62	8,5	-98	2021-07-30 22:17:36
2.607	6,46	65	10	63	6,25	-93	2021-07-30 22:32:38
2.608	6,44	65	10	62	7,5	-93	2021-07-30 22:47:40

**Figura 147**

*Paquetes Perdidos*



## Conclusiones y Trabajos Futuros

### Conclusiones

- Al realizar el análisis de LoRa se concluye que constituye la capa física e incorpora una modulación basada en la técnica CSS, los estándares MQTT y LoRaWAN incorporan todo el concepto de IoT, Nodos, *Gateways*, *Network Servers* y *Application Server* mientras que LoRa se encarga de realizar diferentes procesos como la gestión de la modulación, la potencia y acondicionamiento de la señal de transmisión y recepción. En el proceso de transmisión con LoRa existe la posibilidad de seleccionar seis valores de SF comprendidos entre 7 a 12 con la finalidad de la optimización en cuanto a rendimiento de red y uso del espectro. Con valores de SF bajos, se tiene una transmisión rápida debido a la necesidad de menor número de bits para el envío de información, pero se obtiene un menor alcance.
- Para la elección de topología de red en proyectos con sensores inalámbricos se deben considerar diversos parámetros como la instalación, el costo, flexibilidad, fiabilidad, robustez y consumo energético, etc. En base a la investigación, la topología con repetidores tiene el propósito de mejorar la comunicación con nodos remotos, la topología de malla ofrece la posibilidad de transmitir desde múltiples rutas así como también de transmisiones simultáneas y la posibilidad de escalamiento de red. En el presente proyecto se utilizó la topología en estrella, el *Gateway* LoRa constituye el elemento central que reenvió los mensajes recibidos desde el nodo final hacia el servidor de AWS.
- Se diseñó la red de sensores inalámbricos enfocada a la monitorización de variables físicas en sistemas hidropónicos y se concluye que la tecnología

LPWAN LoRa es apropiada para aplicaciones IoT enfocadas a la automatización de procesos agrícolas debido a que permite el envío ilimitado de mensajes, autonomía en cuanto a la elección del servidor y almacenamiento de información y altos niveles de seguridad al permitir la encriptación y autenticación de cada uno de los Nodos Finales que constituyen la red.

- En el diseño del sistema hidropónico la longitud de los tubos no debe superar los 9 metros o 30 pies, canales de tubería muy largos generan deficiencias en la distribución de los nutrientes, esto debido a que las plantas colocadas en los primeros orificios retienen la mayor cantidad de nutrientes y provocan que las plantas ubicadas en los últimos boquetes del sistema tengan un pésimo desarrollo.
- Se desplegó una red de sensores constituida de un Nodo Final en el sistema hidropónico que permitió la adquisición y procesamiento de los niveles de pH y solución nutritiva lo que hizo posible la producción importante de lechugas en un periodo de 28 días y tomates en 60 días. Asimismo, con la monitorización de temperatura y humedad del suelo fue posible conseguir que las semillas germinen en un tiempo aproximado de 21 días.
- Se desarrolló una interfaz gráfica con lo último en tecnologías front-end y back-end que cumple con el propósito del IoT al recibir y presentar la información al usuario casi en tiempo real y con una baja pérdida de paquetes pues se presenta una pérdida del 2.83%. La plataforma cumple con los estándares de ciberseguridad pues garantiza que los datos se transmitan de manera encriptada al contar con certificados SSL y al combinar Node JS con MQTT la información recibida es almacenada de manera eficiente en una base de datos propia en AWS. Desde la plataforma Web también es posible la

automatización de la bomba para su encendido o apagado por medio del sistema *publish/suscribe* y la inserción de *Set Points* para la rápida emisión de alertas por medio de SMS.

- En las 4 semanas de monitorización se alcanzan valores de RSSI comprendidos entre -80 a -103 dB, mientras que el SNR varía entre 5 a 8.25 dB, lo que permite concluir que se tiene un enlace óptimo, los valores de SNR y RSSI mayores a -7dB y -115 dB respectivamente establecen que el Nodo Final se encuentra instalado cercano al *Gateway* y el entorno no es muy ruidoso. El promedio de retardo en el canal inalámbrico se encuentra en los 124 ms y la pérdida de paquetes cercana al 3% con estos dos valores se garantiza una óptima transmisión al cumplir con los estándares internacionales para aplicaciones en tiempo real con IoT.
- LoRa es una tecnología de bajo consumo enfocado a aplicaciones de IoT, económica e idónea para transmisiones de datos en ambientes en los que se necesiten áreas amplias de cobertura como las zonas agrícolas. Al combinarlo con nuevas técnicas de cultivo como la hidroponía permite la automatización de procesos elementales como riego, controles de temperatura, humedad, pH y mediante la monitorización de las variables fue posible definir que las plantas crecen de manera adecuada en medios levemente ácidos, a medida que proliferan las algas en los tubos del sistema se producen oscilaciones en los niveles de pH y la temperatura adecuada para la eclosión de los huevos de los trips que afecta los tomates ronda entre los 20 a 25 grados.

### **Trabajos Futuros**

- Con el fin de evitar que la instancia y plataforma desarrollada caiga en redes *botnet* o se realice la filtración de información al tener abiertos algunos puertos

en el proceso de transmisión de información se propone implementar herramientas de ciberseguridad, sistema de intrusiones y ataques a plataformas IoT en la nube basado en modelos de Machine Learning.

- Modificar el nodo de monitorización con la incorporación de un sensor que permita realizar la medición de conductividad eléctrica en la solución nutritiva, automatización inteligente para la suministración automática de solución y mezcla de pH, nutrientes y agua por medio de actuadores.
- Se propone el desarrollo e implementación de un sistema de automatización para el control de los niveles de pH que permita tener un rango mucho más ajustado, así como también de un sistema de energía eléctrica que provea de autonomía al sistema en caso de apagones.
- Realizar un análisis exhaustivo de los datos obtenidos de la monitorización, con la finalidad de crear modelos de predicción para la prevención de enfermedades en las hojas y frutos de plantas hidropónicas así como también definir los ambientes de proliferación de algas y larvas insectos por medio de herramientas de machine learning.

## Referencias Bibliográficas

- Alexander, T. (2000). *The best of the growing edge: popular hydroponics and gardening for small commercial growers and hobbyists*. London: New Moon Publishing.
- ARCOTEL. (Diciembre de 2017). *Plan Nacional de Frecuencias Ecuador*. Obtenido de [https://www.arcotel.gob.ec/wp-content/uploads/2018/01/Resolución-12-09-ARCOTEL-2017\\_completa\\_con-firmas-11.pdf](https://www.arcotel.gob.ec/wp-content/uploads/2018/01/Resolución-12-09-ARCOTEL-2017_completa_con-firmas-11.pdf)
- Barani, J. (19 de Abril de 2019). *Barani Design*. Obtenido de <https://www.baranidesign.com/faq-articles/2019/4/23/lorawan-usa-frecuencias-channels-and-sub-bands-for-iot-devices?rq=channels>
- Borrero, J., Fernández, G., & Rodríguez, C. (2018). Aplicación de la tecnología LoRaWAN en la agricultura. *Bo True Activities*.
- Cardona, M., Vijender, K., & Cecilia, G. (2021). *Internet of Medical Things*. Florida: CRC Press.
- Carrasco, G., & Izquierdo, J. (1996). *Técnica de la solución nutritiva recirculante NFT*. Talca: FAO.
- Chaudhari, B., & Zennaro, M. (2020). *LPWAN Technologies for IoT and M2M Applications*. Pune: Academic Press.
- Chaudhari, B., Zennaro, M., & Borkar, S. (2020). LPWAN Technologies: Emerging Application Characteristics, Requirements, and Design Considerations. *Future Internet*, 1-25.
- Citoni, B., Imran, M., & Abassi, Q. (2019). Internet of Things and LoRaWAN-Enabled future Smart Farming. *IEEE*, 14-20.
- Corona, L., Abarca, G., & Mares, J. (2014). *Sensores y Actuadores*. Azcapotzalco: Patria.
- Davcev, D., Mitreski, K., Trajkovic, S., Nikolovski, V., & Koteli, N. (2019). IoT agriculture system based on LoRaWAN. *IEEE*, 1-4.
- Dragino. (2020). *LG01N/OLG01N LoRa Gateway User Manual*. Shenzhen.
- Farshad, F., & Krishnendu, C. (2020). *Intelligent Internet of Things From Device to Fog and Cloud*. New York: Springer.
- Fernández, M., & Uquillas, J. (2020). Análisis de desempeño del estándar LoRaWAN para soluciones de Smart Campus, implementando un sistema de monitoreo IoT en la Universidad de las Fuerzas Armadas - ESPE. *ESPE*, 1-227.
- Ghosly, S. (24 de Marzo de 2017). *All About LoRa and LoRaWAN*. Obtenido de <http://www.sghosly.com>
- Hillar, G. (2017). *MQTT Essentials-A Lightweight IoT Protocol*. Birmingham: Packt.

- Hydro, E. (19 de Noviembre de 2011). *Hydro Environment*. Obtenido de <https://hydroenv.com.mx/catalogo/>
- ITU. (2016). *Datos de propagación para el diseño de sistema de radiodifusión*. Ginebra: ITU.
- Kim, D.-H., Lee, E.-K., & Kim, J. (2019). Experiencing LoRa Network Establishment on a Smart Energy Campus Testbed. *Multidisciplinary Digital Publishing Institute*, 1917.
- Latief, A., & Firasath, N. (2021). *Agriculture 5.0 Artificial Intelligence, IoT and Machine Learning*. Florida: CRC Press.
- Latifa, U., Ramelan, A., & Saputro, J. (2020). Design of Nutrition Automation on Lactuca Sativa NFT Hydroponic Systems. *IEEE*, 645-650.
- Lea, P. (2020). *IoT and Edge Computing for Architects*. Birmingham: Packt.
- Liang, R., Zhao, L., & Wang, P. (2020). Performance Evaluations of LoRa Wireless Communication in Building Environments. *Sensors*.
- LoRa, A. (20 de Febrero de 2020). *LoRa Alliance*. Obtenido de [https://loralliance.org/resource\\_hub/rp2-101-lorawan-regional-parameters-2/](https://loralliance.org/resource_hub/rp2-101-lorawan-regional-parameters-2/)
- Maclay, W. (2017). Comparison of IoT Wireless Standards. *Low Data Rate Links for IoT* (pág. 74). Voler Systems.
- Manzoni, P. (2019). *Intro to MQTT*. Valencia: Universitat Politècnica de Valencia (UPV).
- Marulanda, C., & Izquierdo, J. (2003). *La Huerta Hidropónica Popular*. Santiago de Chile, Chile.
- Mekki, K., Bajic, E., Chaxel, F., & Meyer, F. (2017). A Comparative Study of LPWAN Technologies for Large-scale IoT Deployment. *ICT Express*, 1-7.
- Mekki, K., Bajic, E., Chaxel, F., & Meyer, F. (2019). A Comparative Study of LPWAN Technologies for IoT. *ICT Express*, 5(1), 1-7.
- Milan, M. (2020). *Internet of Things: Concepts and System Design*. Dublin, CA, USA: Springer.
- Mohammed, S. (2018). *Tomorrow's Agriculture "NFT Hydroponics"- Grow within Your Budget*. Switzerland: Springer International .
- Moya, M. (2018). *Evaluación de pasarela LoRa/LoRaWAN en entornos urbanos*. Valencia: Universidad Politécnica de Valencia.
- Murcia, J., & Chacón, L. (2018). Diseño de un sistema automático de cultivo hidropónico para forraje verde. *Universidad de La Salle*, 1-134.
- Noriega, A. (9 de Junio de 2021). *InGenio Learning*. Obtenido de <https://ingenio.edu.pe/aws-vs-azure-vs-google-cual-es-la-mejor-opcion/>

- OMM. (2018). *Guía de instrumentos y métodos de observación*. Ginebra: Organización Meteorológica Mundial.
- Ortiz, G. (2020). *Desarrollo de una red de sensores inalámbricos utilizando tecnología LoRa para el monitoreo de un sistema*. Quito: Universidad Politécnica Salesiana.
- Peng, S.-L., Pal, S., & Huang, L. (2020). *Principles of Internet of Things (IoT) Ecosystem: Insight Paradigm*. Taiwan: Springer.
- Peter, J., & Perttunen, T. (2014). *Network Topologies*. Pune: Sterlite Tech.
- Plastigama. (2018). Tuberías y accesorios de PVC y PE BD. *Plastigama*, 1-28.
- Pol Lim, F. (2016). A Review-Analysis of Network Topologies for Microenterprises. 175-180.
- Quinones, M. (2019). Methodology for the dimensioning and development of an urban LoRaWAN network. *Politecnico Di Torino*, 1-81.
- Quiros, D. (2011). *Diseño de un sistema hidropónico a mediana escala*. Cartago: Instituto Tecnológico de Costa Rica.
- Qusay F, H., Khan, A. u., & Sajjad A, M. (2017). *Internet of Things Challenges, Advances, and Applications*. Florida: Taylor & Francis Group.
- Ramirez, A., & Ruales, C. (2021). *Análisis comparativo de las tecnologías de comunicación para IoT en la monitorización de zonas agrícolas de las provincias de Cotopaxi y Tungurahua*. Sangolquí: Universidad de las Fuerzas Armadas.
- Resh, H. (2004). *Hydroponic food production: a definitive guidebook for the advanced home*. Boca Raton: New Concepts Press Inc.
- Rist, K. (22 de Julio de 2018). *Nordic Semiconductor*. Obtenido de <https://blog.nordicsemi.com/getconnected/iot-wireless-architecture>
- Rodriguez, J., & Kleinschmidt, J. (2020). LoRaWAN Mesh Networks: A Review and Classification of Multihop Communication. *MDPI*, 1-21.
- Sánchez, E. (19 de Enero de 2019). *Aydoagua*. Obtenido de <https://aydoagua.com>
- Seneviratne, P. (2019). *Beginning LoRa Radio Networks With Arduino*. Sri Lanka: Apress.
- Sethi, P., & Sarangi, S. (2017). Internet of Things: Architectures, Protocols, and Applications. *Journal of Electrical and Computer Engineering*, vol. 2017, 25.
- Siranjeevi, R. (28 de Febrero de 2019). *Easydeploy*. Obtenido de <https://www.easydeploy.io/blog/cloud-hosting-price-comparison-chart-aws-azure-google-cloud/>