



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES**

**TEMA:**

**Evaluación de sistemas IoT para monitoreo de variables ambientales  
y su aplicación en la producción agropecuaria**

**AUTOR:**

**Ing. Edwin Fernando Quichimbo Angamarca**

**Trabajo de titulación previo a la obtención del grado de  
Magister en Telecomunicaciones**

**TUTOR:**

**MSc. Manuel Romero Paz**

Guayaquil, a los 11 días del mes de marzo del año 2022



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO**  
**MAESTRÍA EN TELECOMUNICACIONES**

**CERTIFICACIÓN**

Certificamos que el presente trabajo fue realizado en su totalidad por **Edwin Fernando Quichimbo Angamarca** como requerimiento parcial para la obtención del Título de Magíster en Telecomunicaciones.

TUTOR

MSc. Manuel Romero Paz

DIRECTOR DEL PROGRAMA

MSc. Manuel Romero Paz

Guayaquil, a los 11 días del mes de marzo del año 2022



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO**  
**MAESTRÍA EN TELECOMUNICACIONES**  
**DECLARACIÓN DE RESPONSABILIDAD**

Yo, Edwin Fernando Quichimbo Angamarca

DECLARO QUE:

El trabajo de Titulación, **Evaluación de sistemas IoT para monitoreo de variables ambientales y su aplicación en la producción agropecuaria**, previa a la obtención del Título de **Magíster en Telecomunicaciones**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 11 días del mes de marzo del año 2022

EL AUTOR

**Edwin Fernando Quichimbo Angamarca**



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES**

**AUTORIZACIÓN**

Yo, Edwin Fernando Quichimbo Angamarca

Autorizo a la Universidad Católica de Santiago de Guayaquil a la **publicación**, en la biblioteca de la institución del Trabajo de Titulación, **Evaluación de sistemas IoT para monitoreo de variables ambientales y su aplicación en la producción agropecuaria**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 11 días del mes de marzo del año 2022

EL AUTOR

Edwin Fernando Quichimbo Angamarca

# REPORTE URKUND

The screenshot shows the URKUND software interface. At the top left is the URKUND logo. Below it, a table displays document information:

Documento	<a href="#">TT Edwin Quichimbo.docx</a> (D129287227)
Presentado	2022-03-02 20:15 (-05:00)
Presentado por	Luis Córdova Rivadeneira (lcordova@yahoo.com)
Recibido	luis.cordova.ucsg@analysis.orkund.com
Mensaje	TT Ing. Edwin Quichimbo <a href="#">Mostrar el mensaje completo</a>

Below the table, a progress bar indicates that 19% of the document (38 pages) has been processed, with text from 5 sources. The interface includes a toolbar with various icons and a sidebar on the right with a 'Lista de' header.

SISTEMA DE POSGRADO

MAESTRÍA EN TELECOMUNICACIONES

TEMA: EVALUACIÓN DE SISTEMAS IOT PARA MONITOREO DE VARIABLES AMBIENTALES Y SU APLICACIÓN EN LA PRODUCCIÓN AGROPECUARIA

AUTOR: Ing. Edwin Fernando Quichimbo Angamarca

Trabajo de titulación previo a la obtención del grado de Magister en Telecomunicaciones

TUTOR: MSc. Manuel Romero Paz

Guayaquil, a los 03 días del mes marzo del año 2022

SISTEMA DE POSGRADO MAESTRÍA EN TELECOMUNICACIONES

CERTIFICACIÓN Certificamos que el presente trabajo fue realizado en su totalidad por Edwin Fernando Quichimbo Angamarca como requerimiento parcial para la obtención del Título de Magister en Telecomunicaciones.

TUTOR

## **Dedicatoria**

Este trabajo lo dedico a mi familia: a mi esposa Maritza Elizabeth, a mis hijos David Alejandro y Erika Fernanda, razón y fuente de motivación en mi camino. A mis padres Bolívar Vicente y Francisca Justina guías y pilares fundamentales en mi vida. A mi hermano Diego Leonardo, por ser mi apoyo en todo momento.

**Edwin Quichimbo**

## **Agradecimientos**

A Dios por ser esa luz que guía mis pasos y me da la oportunidad de culminar una nueva meta con su bendición. Al MSc. Manuel Romero por sus enseñanzas, guía, tiempo y apoyo en el desarrollo del presente proyecto. A los docentes de la Maestría en Telecomunicaciones de la Universidad Católica de Santiago de Guayaquil a quienes debo gran parte de mis conocimientos.

**Edwin Quichimbo**



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

**SISTEMA DE POSGRADO  
MAESTRÍA EN TELECOMUNICACIONES**

**TRIBUNAL DE SUSTENTACIÓN**

f.

**MSc. Manuel Romero Paz**

TUTOR

f.

**MSc. Manuel Romero Paz**

DIRECTOR DEL PROGRAMA

f.

**MSc. Luis Córdova Rivadeneira**

REVISOR

**MSc. Edgar Quezada Calle**

REVISOR



## RESUMEN

Desde que la medición y monitoreo de variables físicas son de interés para el sector agropecuario, su importancia radica en la gestión de procesos de producción y obtención de indicadores que muestran las fortalezas o debilidades en dichos procesos. En el presente proyecto se realiza un estudio y evaluación a nivel general de la arquitectura para sistemas IoT. Se enfoca en la investigación de soluciones de aplicación en el sector agrícola y ganadero. Se llevará a cabo la implementación de un dashboard en la plataforma IoT Atmosphere y en un servidor de infraestructura en la nube en este caso en Google Cloud en el cual se puedan visualizar mediciones de variables en tiempo real. Las pruebas se realizan a través de la placa de desarrollo PIC-IoT de Microchip, y también a través de software que simula a clientes MQTT, como es el caso de MQTTX. Se procede con el envío de datos hacia plataformas, las cuales muestran en tiempo real las lecturas de las variables monitorizadas. En el desarrollo del presente trabajo se evalúa lo que concierne a los protocolos y tecnologías que conforman una arquitectura funcional para sistema IoT. También se indica el proceso de implementación de determinados servicios en una etapa previa para el desarrollo de una plataforma IoT.

### **Palabras clave:**

IoT, MQTT, Google Cloud, Plataforma Atmosphere, HTTP, JSON, FronEnd, BackEnd, VUE.

## **ABSTRACT**

Since the measurement and monitoring of physical variables are of interest to the agricultural sector, their importance lies in the management of production processes and obtaining indicators that show the strengths or weaknesses in these processes. In this project a study and evaluation at a general level of the architecture for IoT systems is carried out. We focus on the investigation of application solutions in the agricultural and livestock sector. The implementation of a dashboard will be carried out on the IoT Atmosphere platform and on an infrastructure server in the cloud in this case in Google Cloud in which measurements of variables can be viewed in real time. Testing is done through Microchip's PIC-IoT development board, and also through software that simulates MQTT clients, such as MQTTX. Data is sent to platforms which display the readings of the monitored variables in real time. In the development of this work, what concerns the protocols and technologies that make up a functional architecture for an IoT system is evaluated. The process of implementation of services determined in a previous stage for the development of an IoT platform is also indicated.

### **Keywords:**

IoT, MQTT, Google Cloud, Atmosphere Platform, HTTP, JSON, FronEnd, BackEnd, VUE.

ÍNDICE GENERAL	
ÍNDICE DE FIGURAS.....	XIV
ÍNDICE DE TABLAS.....	XV
CAPITULO 1 . Generalidades del proyecto.....	16
1.1 Introducción.....	16
1.2 Antecedentes.....	18
1.3 Definición del problema.....	20
1.4 Justificación.....	20
1.5 Objetivos.....	20
1.5.1 Objetivo General.....	20
1.5.2 Objetivos Específicos.....	20
1.6 Hipótesis.....	21
1.7 Metodología de la investigación.....	21
CAPITULO 2 . Marco Teórico.....	22
2.1 La Internet de las cosas(IoT).....	22
2.2 Evolución de IoT.....	24
2.3 Modelo de referencia de IoT.....	25
2.4 Arquitectura orientadas a servicios de IoT(SOA).....	27
2.4.1 Capa de Objetos(Adquisición de Datos).....	28
2.4.2 Capa de red (Comunicaciones).....	29
2.4.2.1 Redes LAN.....	30
2.4.2.2 Redes WAN.....	30
2.4.3 Capa de servicio (Procesamiento).....	30
2.4.4 Aplicación.....	32
2.5 Modelos de comunicación.....	32
2.5.1 Modelos de comunicación dispositivo a dispositivo.....	33
2.5.2 Modelos de comunicación dispositivo a la nube.....	33
2.5.3 Modelos de comunicación dispositivo a gateway.....	34
2.5.4 Modelos de comunicación a través de Back-End.....	35
2.6 Tecnologías y estándares de comunicación.....	36
2.6.1 Protocolos de comunicación para IoT.....	36
2.6.2 Tecnologías y protocolos de la capa de Objetos (Percepción)	

2.6.2.1	Wifi.....	37
2.6.2.2	Bluetooth.....	38
2.6.2.3	LoRA.....	39
2.6.2.4	SigFox .....	41
2.6.2.5	Zigbee.....	43
2.6.3	Tecnologías y protocolos de la capa de Red .....	44
2.6.3.1	Protocolo IP .....	45
2.6.3.2	Protocolo de internet versión 4 – IPv4.....	46
2.6.3.3	Protocolo de internet versión 6 – IPv6.....	46
2.6.4	Tecnologías y protocolos de la capa de Aplicación .....	48
2.6.4.1	HTTP .....	48
2.6.4.2	MQTT .....	50
2.7	Computación en la nube y plataformas IoT.....	51
CAPITULO 3. Evaluación y desarrollo de plataforma IoT para el		
monitoreo de variables ambientales aplicadas al sector agropecuario.....		
3.1	Aplicaciones IoT en para el sector agropecuario.....	53
3.1.1	Sistemas de gestión de Riego. ....	54
3.1.2	Control de plagas y enfermedades.....	55
3.1.3	Monitoreo de ganado .....	55
3.1.4	Monitoreo de la calidad del agua.....	56
3.1.5	Monitoreo del suelo .....	56
3.1.6	Agricultura de precisión .....	56
3.2	Comparativa de tecnologías de comunicación para IoT.....	57
3.3	Comparativa de servicios en la nube .....	58
3.4	Plataforma IoT Atmosphere .....	60
3.5	Arquitectura IoT general para aplicaciones.....	62
3.6	Implementación de servidor .....	64
3.7	ExpressJS .....	69
3.8	Bases de datos .....	70
3.9	JWT(JSON web token) .....	72
3.10	VueJS.....	73

3.11 Dispositivos IoT .....	74
3.12 Programación de placa IoT .....	75
3.13 Simuladores de clientes MQTT .....	80
CAPÍTULO 4. Plataforma IoT para el monitoreo de variables en el sector agropecuario. ....	82
CONCLUSIONES .....	84
RECOMENDACIONES.....	86
Bibliografía.....	87

## ÍNDICE DE FIGURAS

<b>Figura 2.1.</b> Internet de las Cosas (IoT).....	23
<b>Figura 2.2</b> Evolución de IoT.....	24
<b>Figura 2.3</b> Modelo de referencia de IoT.....	26
<b>Figura 2.4</b> Arquitectura del internet de las Cosas (IoT) .....	27
<b>Figura 2.5</b> Diagrama modelo de comunicación dispositivo a la nube.....	33
<b>Figura 2.6</b> Diagrama modelo de comunicación dispositivo a Gateway .....	34
<b>Figura 2.7</b> Diagrama modelo de comunicación a través de Back – End.....	35
<b>Figura 2.8</b> Arquitectura de red LoRaWan.....	40
<b>Figura 2.9</b> Arquitectura de red SigFox.....	42
<b>Figura 2.10</b> Elementos de red ZigBee .....	43
<b>Figura 2.11</b> Stack de protocolos para IoT .....	48
<b>Figura 2.12</b> Esquema de sistema IoT usando HTTP .....	49
<b>Figura 2.13</b> Secuencia de comunicaciones en HTTP .....	49
<b>Figura 2.14</b> Modelo Publicación - suscripción de MQTT. ....	50
<b>Figura 3.1</b> Sensores más comunes para monitoreo de ganado: 1 Ubicación, 2 Comportamiento de alimentación, 3 Detección de celo, 4 Temperatura, PH, 5 Podómetro, 6,7 Detección de parto.....	55
<b>Figura 3.2</b> Agricultura de precisión.....	57
<b>Figura 3.3</b> Ambiente de desarrollo de Atmosphere para sistemas embebidos..	61
<b>Figura 3.4</b> ambiente de desarrollo para conectividad en la nube Atmosphere ..	61
<b>Figura 3.5</b> Dashboard IoT en la plataforma Atmosphere .....	62
<b>Figura 3.6</b> Topología de arquitectura IoT .....	63
<b>Figura 3.7</b> Creación de instancia en Google Cloud.....	65
<b>Figura 3.8</b> Información de instancia creada en Google Cloud.....	66
<b>Figura 3.9</b> Instalación e inicialización de servidor MQTT.....	67
<b>Figura 3.10</b> Interfaz de Inicio Vue.js .....	74
<b>Figura 3.11</b> Placa PIC-IoT WG Development Board.....	75
<b>Figura 3.12</b> Entorno MCC .....	76
<b>Figura 3.13</b> Placa Pic-IoT en operación .....	80
<b>Figura 3.14</b> Entorno de comunicación EMQx.....	81
<b>Figura 4.1</b> Resultado Dashboard IoT.....	83

## ÍNDICE DE TABLAS

<b>Tabla 2.1</b> <i>Ventajas y desventajas estándar 802.11</i> .....	38
<b>Tabla 2.2</b> <i>Notación IPv4</i> .....	46
<b>Tabla 2.3</b> <i>Notación para direcciones IPv6 en Hexadecimal</i> .....	47
<b>Tabla 3.1</b> <i>Comparación de algunas tecnologías de comunicación para IoT</i> .....	57
<b>Tabla 3.2</b> <i>Comparación de algunas de las plataformas de servicios en la Nube</i>	59

# **CAPITULO 1 . Generalidades del proyecto**

## **1.1 Introducción**

El sector agropecuario tiene una participación importante dentro de los ámbitos que conducen la economía del país, inclusive los gobiernos tanto locales como el nacional promueven varias políticas que buscan impulsar su desarrollo y mejorar las condiciones de este sector, pero pese a todo ello aún debe afrontar múltiples retos marcados por los diferentes cambios que se dan, tanto en lo social, tecnológico, climático etc. Este sector cubre el 95% de la demanda interna de alimentos de la población ecuatoriana (Pino, Aguilar, Apolo, & Sisalema, 2018), es decir, se maneja de forma autosuficiente para satisfacer dicha demanda, aun así, el crecimiento de la población en el país con el tiempo hace que esta demanda también aumente, y para su satisfacción la producción agropecuaria también debe incrementarse, objetivo que solo se puede cumplir si se adoptan mejoras y actualizaciones en los procesos propios de este sector. Mas allá de las políticas públicas en beneficio del sector agropecuario, se hace un enfoque a como los diferentes avances tecnológicos pueden aportar para hacer frente a problemas de baja productividad y rendimiento del sector, hecho que tiene relevancia en el garantizar la seguridad alimentaria de la población ecuatoriana.

Indicados algunos aspectos importantes del sector agropecuario, se pretende analizar desde el punto de vista tecnológico aquellos procesos y actividades que se pueden mejorar con la tecnificación de las mismas, afín de hacer a la tecnología actual una aliada en la mejora de la productividad del sector, cabe aclarar que el presente trabajo se enfoca a dos sub áreas del sector agropecuario, estas son: la ganadería y la agricultura, dejando a un lado los temas de estudio que tengan que ver con acuicultura y pesca.

El internet de las cosas IoT (Internet of Things) constituye un concepto que hace referencia a la conectividad entre objetos físicos, la capacidad de



estos para transferir datos a través de una red y que sea de forma automática. La autonomía y evolución de estos sistemas hacen que surja una nueva revolución en la sociedad, en donde el sector agropecuario no es ajeno al beneficio del uso de las TICs, y en especial las plataformas del internet de las cosas, a través de las cuales se obtiene información de variables medioambientales y comportamientos específicos del sector ganadero. Por citar algunas iniciativas importantes, está la implementación de la denominada agricultura de precisión, que se enfoca a la optimización de recursos que conlleven a la mejora de la productividad y calidad de productos, también se han establecido sistemas de riego autónomos que analizan variables asociadas al suelo, agua y tipos de cultivos; otra área es el análisis de imágenes para detección de plagas en cultivos, collares para el ganado entre otras.

Existen varias tecnologías de comunicaciones, redes y plataformas dedicadas al internet de las cosas, entonces el objetivo que tiene el presente trabajo es vincular el uso de tecnologías IoT con los procesos de producción agrícola y ganadera, esto se consigue analizando el comportamiento de variables presentes en estos campos, obtenida la información correspondiente se la puede procesar y enfocarla a los distintos requerimientos solicitados. También se evaluará las distintas tecnologías tanto a nivel LAN, WAN y plataformas IoT disponibles en la actualidad, a fin de establecer una arquitectura adecuada. Es imperativo realizar un estudio de arquitecturas IoT, componentes usados para monitoreo, adquisición, almacenamiento y tratamiento de datos, todo esto enmarcado dentro de los objetivos de incremento de eficiencia en la producción de sistemas agropecuarios.

Se pretende realizar una simulación de redes de sensores para evaluar el comportamiento y desempeño de este tipo de redes frente a variables medioambientales con vinculación a plataformas de internet de las cosas. Luego de la evaluación se procederá a enunciar recomendaciones que ayuden a impulsar su implementación en el sector agropecuario.

## 1.2 Antecedentes

Los continuos avances en las tecnologías de la información y comunicación han propiciado el surgimiento de sistemas eficientes en el procesamiento y tratamiento de la información, así mismo en el campo de la electrónica se ha logrado reducir el tamaño de los componentes que forman el hardware de dichas tecnologías lo que a su vez permite que exista una convergencia de aquellas actividades cotidianas del mundo real hacía la digitalización del mismas.

Para el presente trabajo de titulación se realiza una investigación de tecnologías actuales dedicadas a realizar mediciones y monitoreo de distintas magnitudes en el mundo real, para ello es necesario empezar con la recolección de información de trabajos similares, además de fuentes provenientes de las distintas organizaciones que ya tienen desarrollados y estandarizados los diferentes protocolos que rigen este tipo de tecnologías. Otro punto de partida es el análisis de los diferentes procesos y magnitudes involucradas en la producción del sector agropecuario, a su vez que estas puedan ser manejadas dentro de variables que sean tratadas como puntos clave para la mejora de la productividad del sector.

El trabajo de (Vite, Vargas, Vargas, & Vargas, 2018) identifica varios casos aplicativos de importancia para el uso del internet de las cosas en las ciencias agropecuarias, que a su vez buscan desarrollar o innovar productos y servicios a fin de que se genere una cultura tecnológica en las áreas productivas, el estudio empieza por identificar las arquitecturas, dispositivos, protocolos y sensores que permitan llevar a cabo la implementación tecnológica basadas en IoT. El estudio de la aplicación de tecnologías IoT se lo hará de forma independiente para el enfoque del sector agrícola y otro para el sector ganadero, en ambos casos se propone algunas soluciones a problemáticas haciendo uso de IoT.

La publicación de la Universidad Técnica de Machala denominada “Análisis de datos agropecuarios” (Ramírez & Mazón, 2018) hace un compendio por cada capítulo con diferentes autores de la ciencia de datos en el sector agropecuario, inicia en el estudio y obtención de estos, el análisis IoT, hasta la aplicación de inteligencia artificial a los datos obtenidos.

Otro de los trabajos de referencia para la presente investigación es el de (Barriga, 2017) mismo que presenta un análisis en el contexto de la interacción humano-máquina e internet de las cosas, a partir de ello establecer una relación sobre aquellas corrientes ideológicas y psicológicas de las actividades humanas donde se puedan intuir nuevas oportunidades de negocio a su vez que analiza el impacto que se tendrá a nivel social, puesto que este tema ya se lo considera como una nueva revolución industrial.

El autor (Coria, 2020) presenta su trabajo en relación a las aplicaciones del internet de las cosas en el mundo real, donde realiza definiciones, plantea las arquitecturas a ser utilizadas y casos en que el IoT es usado para brindar solución a problemáticas presentes en la actualidad.

Por otro lado, el presente trabajo también busca hacer un análisis de las tecnologías y todo lo que esto involucra en materia de aplicación del internet de las cosas a la producción agropecuaria, es decir se debe realizar un estudio de las últimas tecnologías presentes en estos campos, las variables de interés a ser adquiridas y monitorizadas, las plataformas de manejo de información que el sistema recolecte y aquellos resultados que se espera que ocurran de la aplicación de toda esta infraestructura. Puesto que el presente tema involucra un gran campo de acción se recalca el objetivo principal que es el uso de las tecnologías IoT para el mejoramiento de la productividad del sector agropecuario.

### **1.3 Definición del problema**

El problema se plantea como una necesidad de conocer y evaluar las tecnologías basadas en IoT, que sirvan como sistemas de apoyo para mejorar la eficiencia de producción del sector agropecuario.

### **1.4 Justificación**

El sector agropecuario rige su productividad en base a procedimientos comunes y heredados a través de los años, esto hace que la mayoría de los indicadores de producción sean analizados en fases finales de una etapa de productividad sin indicadores claros que permitan tomar las mejores decisiones para mejora de la eficiencia de esta. El crecimiento de la población conlleva a una mayor demanda de productos agropecuarios, que junto con el cambio climático y social obliga a los productores a hacer frente a desafíos mayores, es aquí donde el avance tecnológico, exclusivamente el internet de las cosas permitirá construir sistemas de apoyo inteligentes mejorando la eficiencia de la producción agropecuaria.

### **1.5 Objetivos**

#### **1.5.1 Objetivo General**

Evaluar los sistemas basados en internet de las cosas (IoT), y sus aplicaciones en el monitoreo de variables medioambientales del sector agropecuario que permita mejorar la eficiencia de producción del mismo.

#### **1.5.2 Objetivos Específicos**

1. Analizar el estado actual de las tecnologías usada en el internet de las cosas.
2. Desarrollar una arquitectura basada en red de sensores para el monitoreo y adquisición de datos de variables medioambientales influyentes en el sector agropecuario.
3. Simular una red de sensores para evaluar su desempeño frente a monitoreo de variables medio ambientales.

## **1.6 Hipótesis**

El despliegue de plataformas de internet de las cosas y redes de sensores permitirá automatizar procesos, además permitirá el registro de información en tiempo real la cual coadyuvará a mejorar la producción agropecuaria y promoverá la implementación de sistemas agrícolas inteligentes.

## **1.7 Metodología de la investigación**

El presente trabajo investigativo previo al grado de Magister en Telecomunicaciones es de carácter exploratorio, ya que se exploran las tecnologías usadas para el Internet de las cosas, incluidas la adquisición de señales, arquitecturas para comunicaciones, plataformas de almacenamiento y tratamiento de información etc. Es descriptivo ya que se estudian y se evalúan las diferentes tecnologías IoT como una solución que busca mejorar los procesos de producción agropecuaria.

Además, esta investigación usa el paradigma analítico con un enfoque cuantitativo, ya que se realizan cálculos matemáticos para el diseño de la arquitectura de comunicaciones, y se establecen resultados en tablas y gráficas. Finalmente, el uso de una metodología empírica – analítica a través de una simulación revelará la eficiencia de los modelos propuestos a fin de presentar una topología robusta que permita satisfacer los objetivos planteados para la presente investigación.

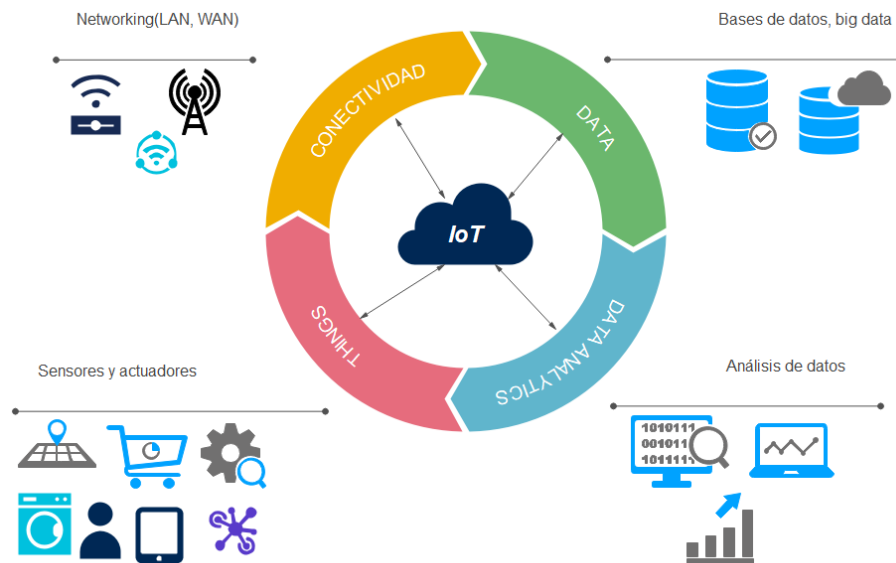
## **CAPITULO 2 . Marco Teórico**

Internet de las cosas (IoT), surge como un paradigma en donde los dispositivos o denominados objetos del mundo real desarrollan su funcionamiento interconectados a través de una red de comunicaciones de datos. Este concepto puede generar un enfoque polémico, toda vez que, así como en la época de la revolución industrial implica también un cambio en la manera cotidiana que se desarrollan las actividades, lo que a su vez puede generar el rechazo de ciertos sectores que verán como amenaza la implementación tecnológica en procesos donde el componente humano deje de ser indispensable. El presente capítulo presenta una descripción de aquellos elementos que rigen el desarrollo del Internet de las cosas, para posteriormente presentar aquellas aplicaciones que resultan de utilidad dentro del sector agropecuario.

### **2.1 La Internet de las cosas (IoT)**

En el año 1999 Kevin Ashton dentro de su trabajo en el campo de la tecnología RFID y la Internet hace uso por primera vez del término Internet de las cosas como aquella visión en donde objetos de la vida cotidiana pudieran ser equipados con tecnología que les permitiera comunicación entre ellos además de poder ser gestionados por computadora. Derivado de este concepto se piensa en las múltiples disciplinas que pueden verse beneficiadas de la aplicación de este concepto, así, por ejemplo, el sector del transporte, el sector de la salud, el sector industrial, el sector agropecuario etc.

La IoT describe la interconexión en red de objetos cotidianos dotados con algún tipo de inteligencia, es decir, consolida una plataforma de dispositivos que se pueden comunicar electrónicamente, comparten información y datos concretos con el entorno que los rodea (Salazar & Silvestre, 2016).



**Figura 2.1.** Internet de las Cosas (IoT)  
**Fuente:** Elaborada por el autor.

Una de las revoluciones de la Internet es la interconexión de personas a través de aplicaciones en dispositivos inteligentes, entonces pensar en una era de conexión de objetos es posible, en donde aquellas actividades desarrolladas manualmente por el ser humano se pueden ejecutar de forma eficiente, esto es derivado del análisis de información y datos involucrados, además, cierto es que los volúmenes de información que resulten a ser procesados serán de considerable tamaño, entonces pensar en su almacenamiento es también motivo de análisis ya que dicha información debe ser posteriormente procesada de la mejor manera para tener una buena interpretación de los datos.

Muchas tecnologías emergen con la Internet de las cosas, así por ejemplo los sistemas de visión artificial, pero la mayoría de estos involucran a sistemas computacionales inteligentes ya sean fijos o móviles que todos los días deberán conectar objetos con sistemas embebidos cuya función no solamente es la de entregar información si no también tener una interacción con el mundo físico.

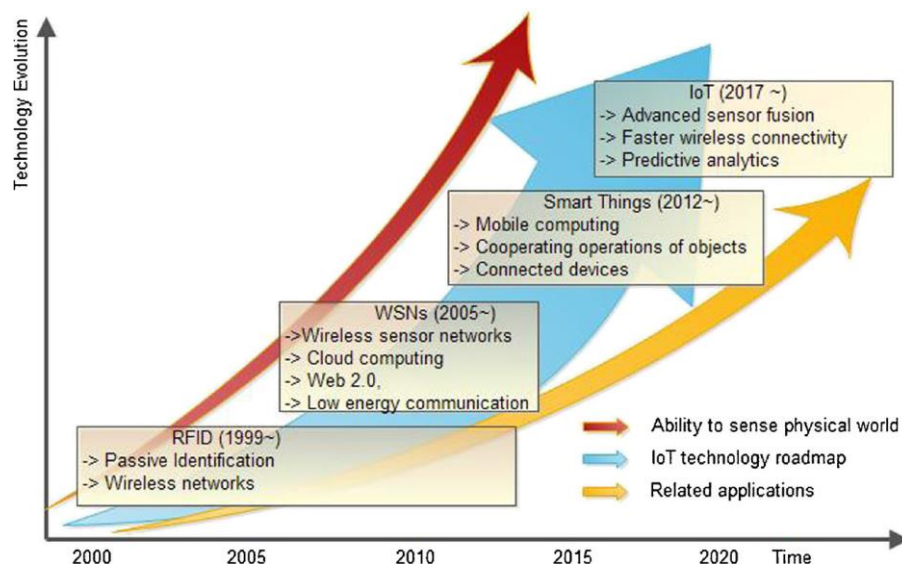
La creación de ambientes inteligentes producto de la interconexión de objetos constituirá una verdadera revolución. De acuerdo con datos

registrados ya desde el año 2011 en número de objetos interconectados a escala mundial superaba al número de personas, para el año 2013 esta cantidad se estima en 9 billones de dispositivos interconectados y para la actualidad esta cifra sigue en aumento que de acuerdo con las estadísticas se indica que está alrededor de los 24 billones de dispositivos conectados (Gubbi, Buyya, Marusic, & Palaniswami, 2013).

Entre algunas de las características que debe poseer un sistema IoT, están la eficiencia, seguridad, escalabilidad, que sea orientado a la computación y almacenamiento de información. Revisando en este grupo de características, también se induce un enfoque en tecnologías de nueva generación, específicamente la computación y almacenamiento en la nube, que, para estos propósitos existen ya lugares especializados en donde la información generada por sensores y plataformas IoT es almacenada e interpretada a fin de que dichos datos sean fáciles de entender además de intuitivos para el usuario final.

## 2.2 Evolución de IoT

La figura 2.2 presenta un esquema del continuo avance que han presentado las diferentes tecnologías para consolidar el internet de las cosas.



**Figura 2.2** Evolución de IoT  
Fuente: (Li, Xu, & Zhao, 2015)



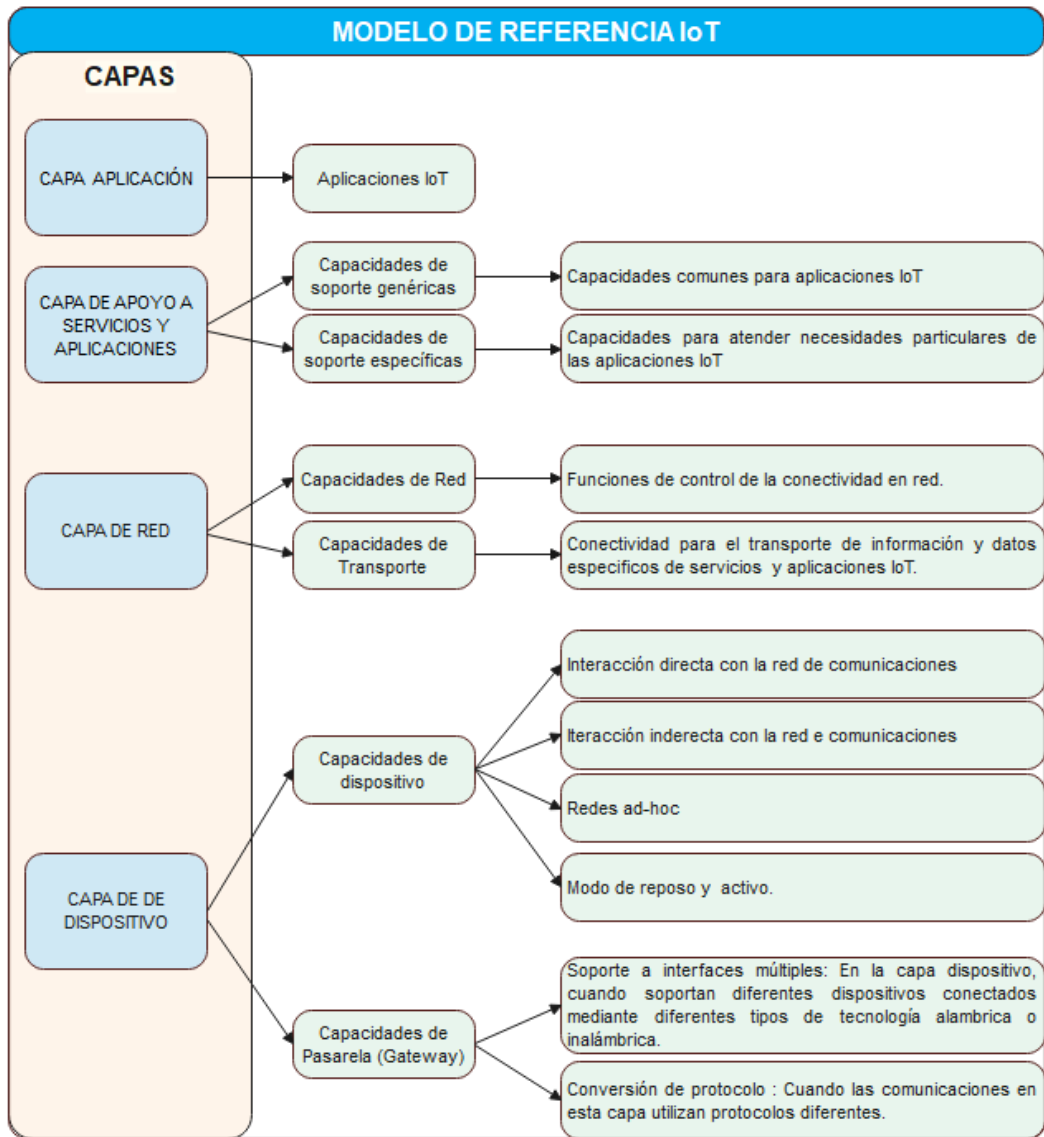
El avance del IoT ha sido posible debido a múltiples factores entre los que destacan la convergencia de la microelectrónica, avances en los sistemas de comunicaciones, electrónica digital y la miniaturización de dispositivos que tienen la capacidad de procesar y comunicar inalámbricamente grandes distancias. Además de ello, la aparición de los servicios en la nube que junto con el Open Source hacen posible que los sistemas IoT estén al alcance de la mayoría de los individuos a un costo prudente y que permitirá la implementación de soluciones haciendo uso de este tipo de tecnología.

El gran volumen de datos que se producen y necesitan ser procesados en tiempo real, requieren aplicar técnicas de Big Data, es decir extraer el valor intrínseco que traen los datos, y de esta información tomar las mejores decisiones para los procesos que se requiera mejorar o automatizar.

Dando un vistazo al trayecto histórico, ya existían formas de llevar a cabo tareas enfocadas al internet de las cosas pero que se realizaban de forma aislada, por lo que la convergencia de estas tecnologías es lo que lleva al éxito de la aplicabilidad de IoT. No se debe dar por terminado la evolución para IoT, ya que con el transcurrir del tiempo seguirán mejorando las tecnologías existentes y posiblemente aparecerán nuevas que deberán ser acopladas y unificadas en las áreas fundamentales del desarrollo de las sociedades.

### **2.3 Modelo de referencia de IoT**

De igual manera la recomendación de la UIT-T Y.2060 (UIT-T, 2012) establece un modelo de referencia para IoT, el cual consta de cuatro capas que interrelacionan capacidades de gestión y seguridad.



**Figura 2.3** Modelo de referencia de IoT  
**Fuente:** Elaborada por el autor basado en (UIT-T, 2012).

De acuerdo con la figura 2.3, las capas que el modelo plantea son: la capa de dispositivo, de red, capa de apoyo a servicios y aplicaciones, y la capa de aplicación. Estas cuatro capas tienen la capacidad de trabajar en función de dotar múltiples aspectos que permitan diseñar, elaborar, implementar, operar y dar seguridad a los sistemas basados en IoT.

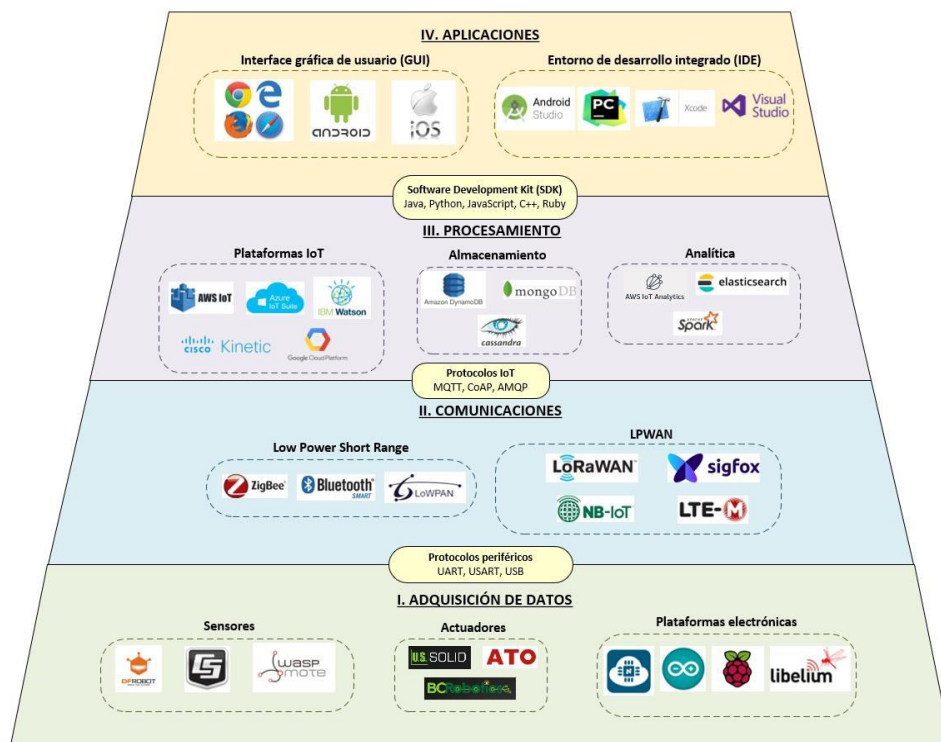
Así mismo, la mencionada recomendación describe algunas características fundamentales que deben ser cumplidas por sistemas IoT, entre las que se mencionan están: la interconectividad, los servicios relacionados con

objetos, heterogeneidad, los cambios dinámicos, y una escala extensa. (UIT-T, 2012)

## 2.4 Arquitectura orientada a servicios de IoT (SOA)

De acuerdo con (Li, Xu, & Zhao, 2015) un requerimiento sine qua non para el funcionamiento IoT es que los objetos en la red estén interconectados, entonces una arquitectura debe garantizar esta interoperabilidad, lo que permite establecer un puente entre las cosas (mundo real) y el mundo virtual.

El diseño de una arquitectura para IoT involucra varios factores tales como el networking, comunicaciones, modelo de negocios, procesos y seguridad. Una arquitectura para IoT debe ser adaptativa y permitir que múltiples dispositivos interactúan unos con otros de forma dinámica además de soportar varias formas de comunicación. Entonces, partiendo del modelo de referencia detallado en la sección 2.3, a continuación, se describe la arquitectura general que un sistema IoT deberá cumplir para su correcta implementación, funcionamiento y operación.



**Figura 2.4** Arquitectura del internet de las Cosas (IoT)  
Fuente: (Zavaleta, 2020)

La figura anterior describe la arquitectura orientada a servicios, misma que consta de 4 capas que comprenden: la capa de objetos, capa de red, capa de servicios y la capa de aplicación.

Cabe señalar que no existe un consenso establecido que indique una arquitectura universal del internet de las cosas, y más bien diferentes arquitecturas son propuestas por los investigadores en función objetivos que se tengan planteados para una determinada solución. Para la presente investigación es aceptada la arquitectura que se indica en la figura 2.4, toda vez se pretende hacer un análisis a nivel global de las tecnologías que permitirán enfocarse en el sector agropecuario.

La arquitectura se la define como un marco de referencia que define los componentes físicos de una red, su organización funcional, configuración, principios de operación, procedimientos y los formatos de datos usados en esta operación. Una arquitectura fundamental para IoT comprende un conjunto de objetos del mundo físico, sensores, servicios en la nube, desarrollo de plataformas, actuadores, capas de comunicaciones, usuarios, modelos de negocios, y por supuesto los protocolos ya establecidos para IoT (Jabraeil, Bahrami, Heidari, Allahverdizadeh, & Norouzi, 2020).

A continuación, se describe las 4 capas que contiene la arquitectura orientada a servicios que actuará como base fundamental para el desarrollo del presente trabajo investigativo.

#### **2.4.1 Capa de Objetos (Adquisición de Datos)**

Constituye la capa más baja de la arquitectura IoT, la cual consiste en una red de sensores, sistemas embebidos, lectores RFID que integran las variables de objetos del mundo real.

Otros autores denominan a esta capa como de percepción en la cual se identifican los objetos y la información se puede obtener a partir de ellos. Dentro de esta percepción se identifica de manera única a los objetos a

través de su dirección física, con ello ya se distingue un entorno en la que existirá transmisión de datos a partir de sensores. Entonces, la percepción constituye la base para la aplicación y desarrollo de IoT (Zhang & Yu, 2013).

Así también de acuerdo con (Li, Xu, & Zhao, 2015) es necesario que los objetos que componen esta capa dispongan de un identificador único digital (UUID) y que sea singular al mismo, esto con el objetivo de rastrearlo y obtener su lectura en el mundo digital. Los autores también indican que a medida que la escala IoT incrementa un gran número de componentes de hardware y software estarán involucrados por lo que se deben considerar aspectos como la eficiencia energética y los protocolos que envuelvan a los sistemas.

En esta capa de la arquitectura IoT se deben considerar algunos aspectos de los sensores y actuadores entre los cuales se mencionan: el costo, tamaño, consumo de energía, la forma de despliegue, comunicaciones e interconexión con la red. Manejar y tener el control de estos aspectos permitirá poder manejar un proyecto de despliegue IoT y evitar que se estancan por situaciones no previstas en el mismo.

#### **2.4.2 Capa de red (Comunicaciones)**

La capa de red es la que conecta todas las things (objetos), y es a través de esta que los objetos pueden compartir datos con una determinada plataforma IoT.

Esta capa es responsable de transferir la información recolectada por los sensores a la siguiente capa. Igualmente se deben tener en cuenta algunos aspectos para esta capa tales como la escalabilidad, flexibilidad, estándares y protocolos para la transferencia de datos. Esta capa debe establecer una red robusta y de alto rendimiento (Naveen, 2016).

Al referirse a la capa de comunicaciones tácitamente se describe lo referente a la función de red y de transporte de acuerdo con el modelo de

referencia IoT. Las tecnologías LAN y WAN deben estudiarse en este campo, ya que comprenden las funciones mencionadas en donde convergen las infraestructuras de redes alámbricas e inalámbricas.

#### **2.4.2.1 Redes LAN**

Denominada como red de área local, su área de cobertura puede ir desde el hogar hasta una red para el ámbito empresarial e industrial.

#### **2.4.2.2 Redes WAN**

Las redes de área amplia constituyen aquellas que permiten comunicar grandes distancias, es decir ciudades, regiones o países. La arquitectura de este tipo de redes no se limita a hacer uso de un solo tipo de tecnología y de proveedor, sino más bien se atraviesa distintas etapas pertenecientes a varias compañías con el fin de lograr el objetivo.

Tanto para las redes LAN como WAN existen diversidad de tecnologías de las cuales es preciso realizar un análisis dependiendo el caso aplicativo para IoT, en la presente se centra en lo que tiene que ver con la agricultura y ganadería, por ende, la evaluación y sugerencia de aplicación de una determinada tecnología será desarrollada más adelante indicando más juicios de valor que justifiquen como optima dicha aplicación. Si bien es cierto, dependiendo la necesidad específica en un ambiente de producción puede manejarse de distinta manera los criterios que aquí se emitan, más bien se busca dotar de un proceso de análisis en función de las capacidades que brindan las tecnologías que se nombren tanto a nivel WAN como LAN. Es por lo que se recalca que la evaluación se efectuará de forma comparativa para que permita visualizar las ventajas y desventajas entre ellas.

#### **2.4.3 Capa de servicio (Procesamiento)**

En esta capa de una arquitectura orientada a servicios se realiza la creación y gestión de servicios requeridos por los usuarios o aplicaciones de software. Esta capa se basa en la tecnología middleware, misma que es

primordial para consumir servicios y ejecución de aplicaciones de IoT (Sosa, Tello, & Lara).

Esta capa constituye una de las capas más críticas de la arquitectura que opera en modo bidireccional. Esta actúa como una interfaz entre la capa de objetos y la capa de aplicación. Así mismo esta capa es responsable de funciones críticas tales como la administración de dispositivos, administración de información, agregación de datos, análisis semántico, control de acceso, y descubrimiento de información (Bandyopadhyay & Sen, 2011).

Middleware describe a un sistema que brinda funciones y servicios comunes para aplicaciones, este se encarga de las tareas de gestión de datos, los servicios para las aplicaciones, mensajería, autenticación y gestión de API (RedHat, 2018). Entonces se dice que esto permitirá interactuar con los elementos inherentes a un ordenador, ya sea al sistema operativo, hardware o con aplicaciones.

Así también, de acuerdo con (Arroyo, 2019), la tecnología middleware actúa como una capa de integración para sensores, actuadores, dispositivos y aplicaciones. Tiene la responsabilidad de tomar datos de los dispositivos conectados, procesarlos, suministrar datos a las aplicaciones conectadas, y de los dispositivos de control. Los dispositivos pueden establecer una comunicación directa con el Middleware, para lo debe admitirse una tecnología adecuada de comunicación, transporte y carga.

En esta capa de la arquitectura de una plataforma IoT se lleva a cabo lo relativo al análisis de datos, y cuando se realiza a gran escala ya se refiere a lo que es Big Data, a más de ello actualmente el modelo de computación en la nube surge como un modelo de negocio para las arquitectura IoT, inclusive en la actualidad grandes empresas a nivel mundial brindan sus servicios basados en este paradigma en pos de una mejora a la competitividad y como alternativa viable de solución a múltiples proyectos

que nacen día tras día. Así como en las tecnologías de networking, es necesario realizar una evaluación a los servicios disponibles en la nube y a su vez permita vislumbrar las características que debe cumplir una plataforma enfocada para IoT en el sector agropecuario. En el Capítulo 3 se evalúan algunas de las principales plataformas de computación en la nube, recalcando que no necesariamente estas son ineludibles, ya que en avance de este tipo de servicios día a día surgen multitud de estos, mejorando así la competitividad.

#### **2.4.4 Aplicación**

La capa de aplicación realiza la entrega de las aplicaciones a los diferentes usuarios de IoT, esta capa proporciona diversas funciones tales como almacenar, organizar procesar y compartir el entorno de datos obtenidos por parte de los sensores. La capa de aplicación es el nivel superior y representa la última tarea para la gestión y planificación, que a su vez requiere una gran capacidad computacional.

#### **2.5 Modelos de comunicación**

De acuerdo con la RFC 7452 (Tschofenig, Arkko, Thaler, & McPherson, 2015) se presentan cuatro modelos de comunicación utilizados en ambientes de objetos inteligentes, de estos es posible que más de un modelo pueda aplicarse a una solución IoT, además se otorga flexibilidad para que los desarrolladores reutilicen los modelos basados en experiencias de otros sectores.

El despliegue de dispositivos IoT en el futuro será de considerables magnitudes que a su vez consolidará una inteligencia para múltiples sectores. Conocer los diversos modelos de comunicación para IoT proporciona un gran valor, no solo para desarrolladores, sino más bien para múltiples actores involucrados hasta el usuario final que se verá beneficiado de modelos de comunicación efectivos que a su vez abrirán oportunidades para el crecimiento del sector que se aplique.



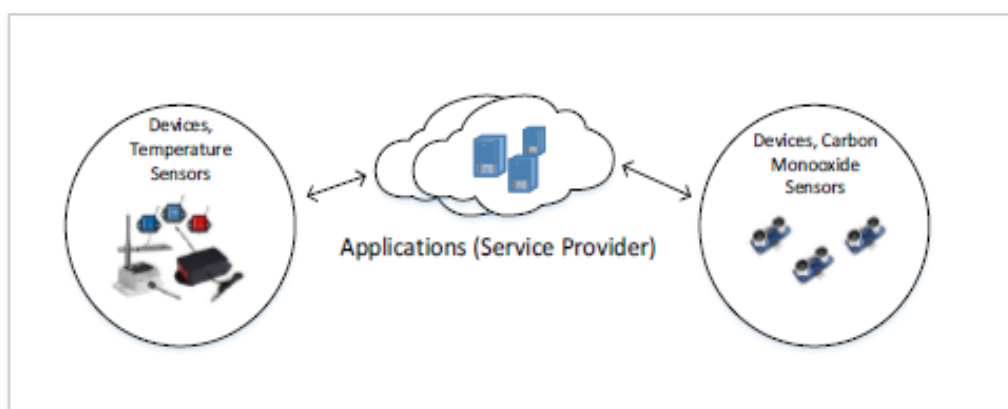
### 2.5.1 Modelos de comunicación dispositivo a dispositivo

En este modelo los dispositivos sin importar su fabricante pueden interoperar y comunicarse directamente, esta comunicación se realiza en tiempo real para transmitir, recibir o desencadenar una determinada acción en el sistema IoT.

De igual manera, según menciona (Kulkarni & Kulkarni, 2017) en este modelo dos o más dispositivos se conectan y comunican directamente entre sí. La comunicación se puede llevar a cabo a través de múltiples tipos y topologías de redes incluidas las redes IP. Los protocolos que generalmente se utilizan son Bluetooth, Z-Wave o ZigBee, ya que se caracterizan por tener una baja tasa de transmisión de datos y optimización de fuentes de energía a los dispositivos.

### 2.5.2 Modelos de comunicación dispositivo a la nube

En este modelo los dispositivos IoT se conectan directamente a un servicio de nube en internet, mismo que es capaz de intercambiar y procesar datos de una determinada solución IoT. En este modelo se aprovechan las infraestructuras de comunicación tradicionales que generalmente ya existen en ambientes de producción, ya sean están cableadas como Ethernet o inalámbricas por Wi-Fi, de esta manera se establece la conexión entre el dispositivo y una red IP, que finalmente se conectará a los servicios requeridos y que se ejecutan en la nube (Aliero, Ahmad, Kalgo, & Aliero, 2020).



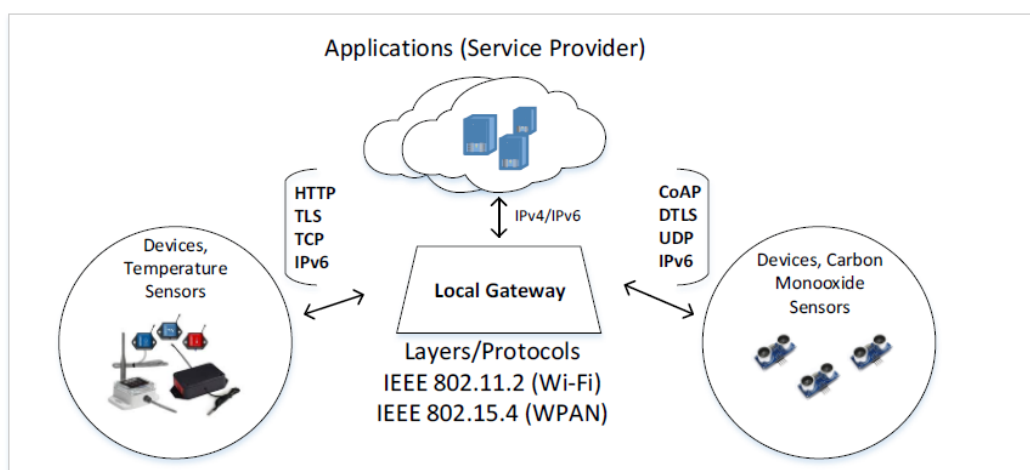
**Figura 2.5** Diagrama modelo de comunicación dispositivo a la nube  
**Fuente:** (Aliero, Ahmad, Kalgo, & Aliero, 2020)

Si bien es cierto este modelo brinda las ventajas de tomar la infraestructura de una red IP fiable en producción y permite conectarse a los servicios en la nube, existe el caso en que empresas dedicadas a este tipo de servicios realizan la fabricación de dispositivos y a su vez promueven el uso de protocolos de comunicación propios, lo que hace que no exista compatibilidad entre fabricantes, limitando así el uso de dispositivos de diferentes fabricantes. Por ello es necesario que se cumpla el que exista una estandarización a fin de que, sin importar la empresa que fabrica un determinado dispositivo los servicios en la nube puedan integrarse a las diversas plataformas en la nube que día a día surgen.

El usuario es capaz de controlar el dispositivo remotamente a través de comandos que pueden ser enviados desde un aplicativo que generalmente es web, ya sea desde un celular, Tablet o computadora.

### 2.5.3 Modelos de comunicación dispositivo a gateway

El modelo de dispositivo a Gateway o puerta de enlace emplea un dispositivo que ejecuta un software de aplicación, que le permite actuar como intermediario entre el dispositivo IoT y el servicio en la nube, así mismo debe garantizar seguridad y otras funciones que garanticen la interoperabilidad de los dispositivos involucrados.

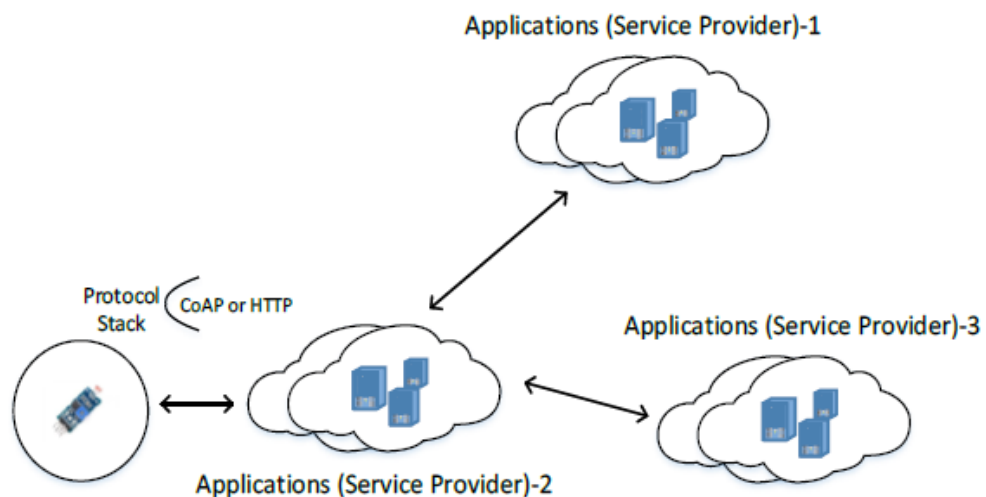


**Figura 2.6** Diagrama modelo de comunicación dispositivo a Gateway  
**Fuente:** (Aliero, Ahmad, Kalgo, & Aliero, 2020)

Una considerable cantidad de dispositivos funcionan en base a este modelo. En muchos de los casos el dispositivo que hace de Gateway es un smartphone, que corre en su sistema una aplicación que tiene la capacidad de comunicarse con los servicios en la nube. Como ejemplo de este modelo están los dispositivos usados para el monitoreo de actividad deportiva, que usan un smartphone para transmitir sus datos a un servicio en la nube ya que los mismos no tienen la capacidad de conectarse directamente a dichos servicios (Aliero, Ahmad, Kalgo, & Aliero, 2020).

#### 2.5.4 Modelos de comunicación a través de Back-End

Este modelo se refiere a una arquitectura de comunicación que posibilita a los usuarios que los datos de objetos inteligentes puedan ser exportados y analizados desde un servicio en la nube en combinación con datos de otras fuentes.



**Figura 2.7** Diagrama modelo de comunicación a través de Back – End  
**Fuente:** (Aliero, Ahmad, Kalgo, & Aliero, 2020)

Este tipo de arquitectura permite a los usuarios otorgar acceso de los datos de sensores a terceros, lo que a su vez brinda la opción de no depender de sitios únicos para almacenar los datos de dispositivos, y más bien si el usuario requiere cambiar de servicio IoT puede trasladar sus datos según sea su necesidad. Este enfoque es una extensión del modelo de comunicación de dispositivo único a la nube (Aliero, Ahmad, Kalgo, & Aliero, 2020).

Al referirse a Back – End, (posteriormente se entrará en más detalles), se define como una tecnología presente en la parte de los servidores, que permite interactuar con bases de datos, procesar solicitudes, generar y enviar respuestas al cliente a través de un Front – End. La presente investigación pretende implementar tanto a nivel de Front y Back – End una plataforma IoT que permita ejecutar lo descrito anteriormente, y su vez, para lograr este propósito se deben conocer varios aspectos los cuales se describirá en el desarrollo de la presente tesis.

Back – End requiere de interfaces de programación de aplicaciones (Apis) en la nube, para permitir dentro de su modelo el intercambio de datos e interoperabilidad de estos en la nube.

## **2.6 Tecnologías y estándares de comunicación**

En esta sección se describe las principales tecnologías empleadas en el campo del internet de las cosas, así como también los protocolos y estándares principalmente usados en las capas del modelo de referencia que se ha analizado. Esto permitirá definir cuáles son las aplicables dentro del campo de interés de esta investigación (el sector agropecuario), y de igual manera aclarará todas las nociones para el desarrollo de una plataforma IoT en lo que tiene que ver con la recolección, transmisión, procesamiento, y visualización de datos correspondientes a variables ambientales u otras que se requiera su análisis.

### **2.6.1 Protocolos de comunicación para IoT**

Al referirse al término protocolo dentro de comunicaciones en redes de datos se establece como el conjunto de reglas y normas necesarias para garantizar una comunicación confiable y eficiente entre nodos, que en IoT corresponde a los dispositivos y equipos involucrados.

El análisis de los protocolos para IoT se hará en base al modelo de referencia estudiado, y si bien es cierto para las capas de dispositivos y red (internet) ya son de conocimiento general aplicables una gran cantidad de

aplicaciones de redes IP, en la capa de aplicación surgen nuevos protocolos que han sido creados con el fin de dar solución y enfocarse de manera exclusiva en IoT, pero en todo caso se describirá de forma concreta aquellos protocolos y tecnologías que sean de interés para la llevar a cabo los objetivos planteados en la presente investigación.

### **2.6.2 Tecnologías y protocolos de la capa de Objetos (Percepción)**

Las redes de comunicaciones constituyen uno de los aspectos principales para el desarrollo de proyectos basados en Internet de las cosas, a través de estas es que se puede conectar los dispositivos involucrados. La evolución y desarrollo tecnológico permite que múltiples tecnologías de comunicaciones emerjan y se posicionen como las más idóneas para la conectividad de dispositivos y transmisión de datos que estos generen prácticamente desde cualquier lugar geográfico. En IoT son varias tecnologías que se tendrá a disposición para su uso en la comunicación de dispositivos, esto en función de múltiples factores que se analicen y consideren en la etapa de desarrollo. A continuación, se indican algunas de las principales tecnologías que se disponen en la actualidad para las comunicaciones en el internet de las cosas.

#### **2.6.2.1 Wifi**

WIFI constituye una tecnología regida por el estándar IEEE 802.11, tiene asignadas las bandas ISM (Industrial, Scientific and Medical), para uso en redes inalámbricas que funcionaran como redes de área local.

El estándar 802.11 establece los protocolos que garantizan la compatibilidad e interoperabilidad de los dispositivos certificados bajo este estándar. Wifi usa CSMA/CA en la capa de acceso al medio a diferencia de ethernet que usa CSMA/CD. Wifi maneja grandes anchos de banda a un precio reducido, lo que lo hace unas de las mejores opciones para transmitir datos en redes IP. También se puede utilizar cumpliendo ciertas condiciones y normas legales en ambientes externos, en los cuales se logra comunicaciones de varias decenas de kilómetros en topología punto a punto (Camacho, et al., 2008).

En la actualidad existen varios tipos de comunicación que establece el estándar 802.11, haciendo uso de espectro no licenciado a 2.4 y 5 GHz. Dentro de estas familias están: 802.11n que transmite 11Mbps, el 802.11g cuya velocidad está en lo 54Mbps, y ahora uno de los más populares el estándar 802.11n que puede operar en ambas bandas (2.4 y 5 GHz) con velocidades teóricas de 400 a 600 Mbps (Song & Issac, 2014-11). También está el estándar 802.11ac que opera únicamente en la banda de 5GHz, y este trabaja a una velocidad máxima de 1300 Mbps.

**Tabla 2.1** Ventajas y desventajas estándar 802.11

Ventajas	Desventajas
<ul style="list-style-type: none"> <li>– Uso de frecuencia en bandas no licenciadas (ISM).</li> <li>– Altas velocidades de trabajo.</li> <li>– Estandar ampliamente difundido y fácil de configurar.</li> <li>– Bajo consumo de potencia.</li> <li>– Flexibilidad en topologías.</li> </ul>	<ul style="list-style-type: none"> <li>– Interferencias de portadoras al trabajar en bandas no licenciadas.</li> <li>– Tecnología enfocada a redes inalámbricas de corto alcance.</li> <li>– El número de colisiones aumenta proporcionalmente en relacion al número de usuarios.</li> <li>– Baja capacidad de penetración.</li> </ul>

**Fuente:** (Camacho, et al., 2008)

De las topologías existentes, la denominada Punto a Multipunto será la que se usará, es decir los dispositivos establecerán comunicación con un punto de acceso centralizado común, mismo que gestionara el tráfico de datos que genere la red inalámbrica como tal.

### **2.6.2.2 Bluetooth**

Constituye una tecnología de comunicaciones inalámbricas de corto alcance que reemplaza cables que conectan los dispositivos electrónicos. El transceptor de RF Bluetooth trabaja en la banda ISM de 2.4 GHz, y utiliza un transceptor de salto de frecuencia para combatir la interferencia.

Bluetooth administra los dispositivos basado en la topología tipo estrella, ha este grupo de dispositivos sincronizados se lo conoce como piconet, mismo que puede contener un maestro y hasta siete esclavos activos. En un piconet, el canal de radio físico es compartido por un grupo de dispositivos sincronizados por un reloj común y saltos de frecuencia (Scientific American, 2007).

Para que Bluetooth sea una opción viable para IoT, este requiere tener un bajo consumo de energía, de tal manera que las baterías de los dispositivos tengas largos periodos de alimentación y vida para los dispositivos que hagan uso de esta tecnología. Es así que un nuevo estándar aparece en la actualidad, denominado “Bluetooth Low Energy” (introducido por Special Interest Group SIG), el cual busca optimizar y perfeccionar la operatividad entre los dispositivos para IoT, que a su vez hace a esta tecnología fiable y segura para la interconexión de varios dispositivos.

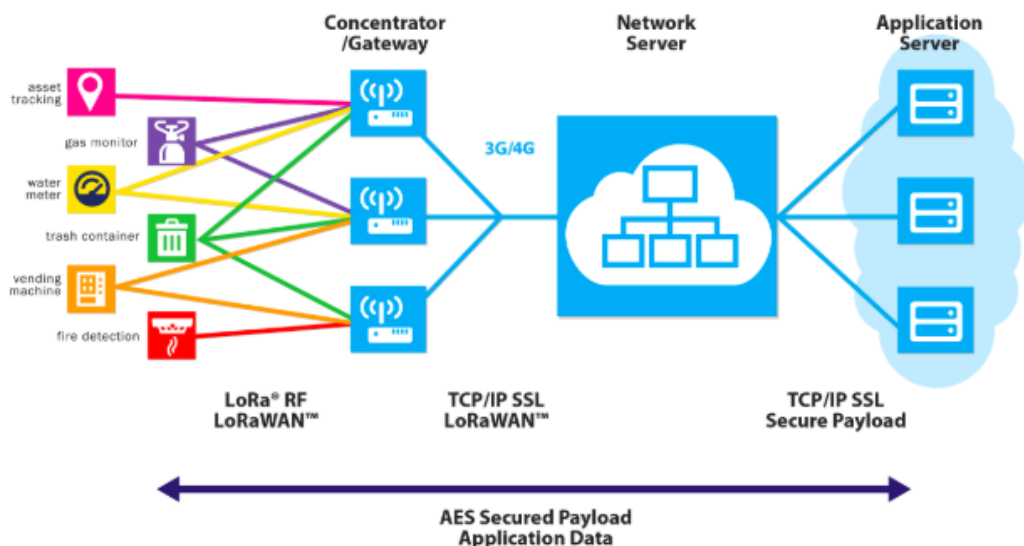
### **2.6.2.3 LoRA**

Hasta ahora las tecnologías mencionadas establecen redes inalámbricas de área local (LAN) de relativo corto alcance, pero existirá el caso de aplicaciones que requieran alta eficiencia en el consumo energético y establecer comunicaciones a través de grandes distancias. Como ejemplo se menciona sistemas inteligentes de posicionamiento, agricultura inteligente entre otros. Es aquí donde existe una tendencia y prevalencia de la tecnología LoRa (Long Range), que puede satisfacer los requerimientos solicitados, es decir LoRa es la mejor opción para aplicaciones IoT las cuales requieran grandes distancias de comunicación mientras hace muy poco uso de la fuente de energía del dispositivo (Zourmand, Kun Hing, Wai Hung, & AbdulRehman, 2019).

La tecnología LoRa ofrece seguridad en la transmisión de datos, hace uso de la banda ISM para la transmisión en radiofrecuencia, y usa técnicas de modulación de espectro ensanchado conocida como Chirp Spread Spectrum (CSS) de características parecidas a la modulación FSK, este tipo de modulación la hace de gran tolerancia frente a interferencias.

LoRaWAN, define el protocolo de comunicación y la arquitectura del sistema, mientras que LoRa define la capa física, así mismo según lo indica (Khutsoane, Isong, & Abu-Mahfouz, 2017) LoRaWan constituye un protocolo de comunicación inalámbrico desarrollado por la Alianza LoRa, quienes se enfocaron en comunicaciones de largo alcance con visión para IoT. En conclusión, LoRaWan es el protocolo encargado de unir múltiples dispositivos LoRa, y a su vez gestionar sus canales de comunicación, los parámetros sobre los cuales se realiza la conexión, anchos de banda, cifrado, etc. En América, las bandas de frecuencia que utiliza LoRa son: 433 MHz, 868 MHz y 915 MHz (Coria, 2020).

Las redes LoRa hacen uso de la topología tipo estrella, donde cada nodo está en la capacidad de enviar mensajes hacia varios Gateway que se formen parte de la red. Los Gateway mantendrán conexión con servidores de Red a través de enlace IP. También es necesario mencionar que desde un dispositivo o nodo final no se puede enviar mensajes a un determinado Gateway y más bien los mensajes que se generan desde los dispositivos son recibidos por todos los Gateways en la zona de cobertura (Coria, 2020).



**Figura 2.8** Arquitectura de red LoRaWan  
**Fuente:** (Coria, 2020)



De la figura 2.8 se puede identificar los siguientes elementos que conforman una red LoRa:

- **Dispositivos Finales:** constituyen los dispositivos de la capa de percepción, que generalmente son sensores que envían información constante a los gateways. Alcanzan grandes distancias de transmisión enviando pequeñas cantidades de datos.
- **Gateway LoRa:** Este recibe los paquetes que se van desde los dispositivos finales, para posteriormente a través de enlaces IP u otro tipo de conexión enviarlos al servidor de Red.
- **Servidor de Red:** Es el encargado de gestionar toda la red como tal. Al momento de recibir un paquete, elimina redundancias, verifica los protocolos de seguridad e integridad del mensaje, para enviar una respuesta de acknowledge al Gateway mejor posicionado.
- **Servidor de aplicación:** En este servidor se procesan todos los datos enviados desde los dispositivos finales y se toman acciones en base a la información y aplicación de que se haya desarrollado.

#### 2.6.2.4 SigFox

Otra de las tecnologías que está ganando campo en el desarrollo de soluciones IoT es SigFox, clasificada dentro de las tecnologías LPWAN, siendo una de las más representativas en este campo. Fue creada en 2009 por Ludovic Le Moan y Christophe Fournier en Francia. Esta proporciona de forma rentable conectividad entre los dispositivos y la nube. La tecnología SigFox tiene la capacidad de conectar gran cantidad de dispositivos, inclusive se habla de miles de millones, ya que el enfoque que maneja es conectar los dispositivos a una infraestructura de nube central y donde no existe la necesidad de servicios de roaming para los dispositivos de usuario final (Hemjal, 2019). El mismo autor indica que cada estación base SigFox puede manejar hasta tres millones de dispositivos, cuya gestión es sencilla de manejar y si se requiere aumentar la cantidad de estos se lo puede conseguir aumentando la cantidad de estaciones base. Así mismo afirma que SigFox posee un buen rendimiento tanto en exteriores como en

interiores, y en términos de porcentajes para fallos para esta tecnología se establece solo en el 12%.

Una celda SigFox puede proporcionar una cobertura aproximadamente de 30 – 50 Km en áreas rurales y 3 – 10 en áreas urbanas. La tecnología se basa en Low Throughput Network (LTN), que entre las tecnologías LPWA admiten la transmisión de pequeñas cantidades de datos a través de grandes distancias, y también pueden coexistir con tecnologías inalámbricas celulares como LTE, GSM, CDMA, entre otras. Pensando en la elevada cantidad de dispositivos que se podrían conectar a estas redes, y para equilibrar el ancho de banda en la comunicación y con ello evitar que se creen colisiones y retardos en el procesamiento de mensajes, SigFox permite transferir 140 mensajes, cada uno de 12 bytes de longitud, en 245 horas con un límite máximo de 7 mensajes por hora (Joseph, 2018).



**Figura 2.9** Arquitectura de red SigFox  
**Fuente:** (Joseph, 2018)

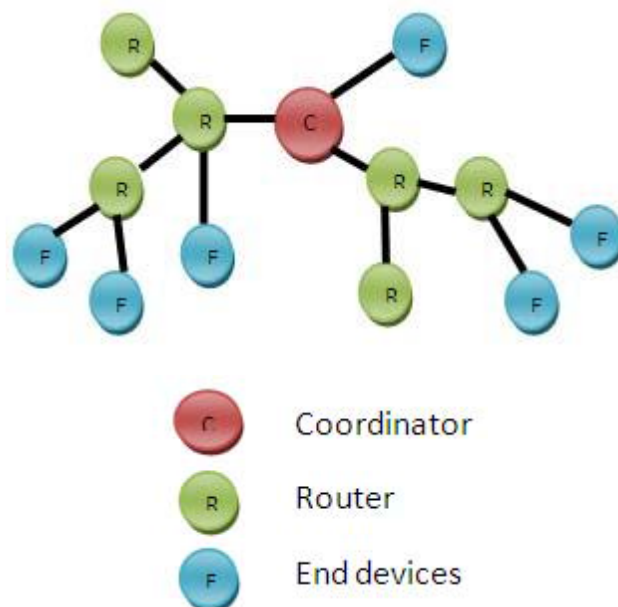
La figura 2.9 describe la arquitectura típica de una red SigFox, esta es similar a una red telefónica celular, cuenta con estaciones receptoras que reciben los mensajes y los envían a las estaciones bases. Los dispositivos de usuario suelen ser de tamaño pequeño con sensores capaces de transmitir información a las antenas receptoras de las estaciones, que luego transmiten el mensaje a la nube SigFox, esta corresponde a una plataforma que permite manejar los dispositivos y visualizar los mensajes de cada uno de ellos. SigFox trabaja con bandas no licenciadas y constituye una red

pública con lo que está sujeta a cumplir con reglas y normas comunes a todos los usuarios (Coria, 2020).

### 2.6.2.5 Zigbee

La tecnología Zigbee está basada en el protocolo IEEE802.15.4, fue creada por la Zigbee Alliance como una tecnología de comunicación inalámbrica de corto alcance en rangos de 10 a 15 m. Trabaja en la banda de 2.4 GHz, admite velocidades de 250 kbits/s en los canales. También opera a 858 MHz para Europa y 915 MHz para Estado Unidos y Australia, esta hace usos de técnicas de modulación de espectro ensanchado por secuencia directa (DSSS). Esta tecnología está pensada para soluciones que requieran bajo consumo energético, incluidos domótica y automatización (Mora & Urrego, 2018).

La energía necesaria para ZigBee es muy pequeña, en la mayor parte de los casos usa 1mW o incluso menos. Pese a esto ofrece un alcance de hasta 150 m en exteriores, maneja velocidades de 20 kbps, 40 kbps y 250 kbps (Ramya, Shanmugaraj, & Prabakaran, 2011).



**Figura 2.10** Elementos de red ZigBee  
**Fuente:** (Ramya, Shanmugaraj, & Prabakaran, 2011)

En la figura 2.10 se indican los elementos que conforman una red ZigBee, lo cual se detalla a continuación:

**Coordinador:** Es el encargado de controlar la comunicación en la red, esto se refiere a iniciar la red, seleccionar los parámetros de esta para su funcionamiento tales como el canal de radiofrecuencia, identificador de red único además de establecer otros parámetros operativos. También puede almacenar información de red y claves de seguridad, por todo esto debe de poseer características adecuadas para el procesamiento de datos (Ramya, Shanmugaraj, & Prabakaran, 2011).

**Router:** Este actúa como nodo intermedio, retransmitiendo datos de otros dispositivos. El router puede conectarse a una red existente, es capaz de aceptar conexiones de otros dispositivos y actuar como retransmisor de red. La red se puede ampliar a través del uso de enrutadores ZigBee (Ramya, Shanmugaraj, & Prabakaran, 2011).

**Dispositivo Final:** Estos dispositivos pueden establecer comunicación con el nodo padre, que puede ser el coordinador o el router, pero no pueden transmitir datos desde otros dispositivos. El dispositivo final puede ser de bajo consumo, recolecta información de varios sensores. Estos dispositivos no tienen que permanecer activos todo el tiempo, a diferencia de los de otras categorías que sí deben hacerlo (Ramya, Shanmugaraj, & Prabakaran, 2011).

En Zigbee la comunicación puede establecerse en topología estrella, entre pares, árbol de grupos o a su vez en topología malla.

### **2.6.3 Tecnologías y protocolos de la capa de Red**

La capa de red en el campo del networking es la que se encarga de llevar a cabo procesos y servicios para la transferencia de datos de extremo a extremo a través de la red entre dispositivos. A continuación, se detallan características del protocolo IP (IPv4 e IPv6), mismo que constituye el

protocolo de transporte de paquetes de datos de la capa 3 ampliamente utilizado en las redes de comunicaciones.

### **2.6.3.1 Protocolo IP**

El protocolo de internet está diseñado para uso en sistemas interconectados de redes de comunicación de conmutación de paquetes. Este protocolo permite la transmisión de bloques de datos desde un origen a un destino, en donde estos son claramente identificados por direcciones de longitud fija. El protocolo IP también permite fragmentación y reensamblaje de datagramas que son transmitidos a través de las redes (Postel, 1981)

IP maneja en enrutamiento de datos entre redes y hosts, además al definir un esquema de direcciones, IP maneja la transmisión de datos desde una computadora de origen a una especificada por la dirección IP de destino, esto lo consigue dividiendo grandes cantidades de datos en un paquete IP fácilmente administrable que se puede entregar a través de la red. Para administrar mejor el tráfico de la red, IP especifica el protocolo para dividir mensajes individuales en una gran cantidad de porciones, donde cada una de ellas es responsable de encontrar su camino a través de la red en función de los cambios, congestión etc. Cada vez que llega un mensaje a un enrutador IP, este decide donde enviarlo a continuación. Los enrutadores pueden enviar datos a lo largo del camino de menor resistencia independientemente del tráfico de la red local (Curran, 2009).

El protocolo IP presenta las siguientes características:

- Fragmentación de paquetes de ser necesario.
- El direccionamiento lo maneja a través de direcciones lógicas IP de 32 bits (IPv4) y 128 bits (IPv6)
- En caso de que un datagrama no sea entregado, este permanecerá en la red por un tiempo finito.

Para establecer una conexión a internet, todo dispositivo tiene que poseer una dirección IP, esta tiene que ser única e irrepetible asignada a cada interfaz del dispositivo. En la actualidad existen dos versiones o formatos de direcciones IPv4 e IPv6.

### 2.6.3.2 Protocolo de internet versión 4 – IPv4

Estas direcciones incluyen un sistema de direccionamiento con identificadores numéricos de 32 bits, esta longitud limita el número de total de direcciones posibles a 4.294.967.296 ( $2^{32}$ ) en todo el mundo, y es la versión más usada en la actualidad.

Los 32 bits usualmente se escriben utilizando la notación decimal con punto. Al referirse a decimal, es por el hecho de que cada byte (8 bits) de la dirección IP de 32 bits se muestra mediante su equivalente en decimal, ósea los cuatro números decimales se escriben uno detrás de otro con puntos decimales como separadores. (Odom, 2014)

**Tabla 2.2** Notación IPv4

Notación decimal	Notación binaria
192.168.4.16	11000000 10101000 10101000 00010000
10.50.34.5	00001010 00110010 00100010 00000101

**Fuente:** *Elaboración propia*

Puesto que cada octeto representa un número binario de 8 bits, el rango de números decimales de cada octeto va de 0 a 255.

### 2.6.3.3 Protocolo de internet versión 6 – IPv6

El continuo incremento de dispositivos a internet desencadenó un problema para IPv4, el de no poder abastecer la demanda de direcciones y por ende el impedimento de crecimiento de la red. IPv6 surge para sustituir a IPv4 y dar solución a esta problemática de direcciones para equipos que se

pueden conectar a internet. IPv6 incrementa el número de direcciones al utilizar 128 bits.

IPv6, al utilizar un campo de dirección diferente, además de algunas de direccionamiento distintas, implica que también otros protocolos y funciones cambien, pero el objetivo principal es el mismo que el del protocolo IPv4. Así como se lo define en la RFC 2460, IPv6 define un concepto de paquete, de direcciones para dichos paquetes y el rol para los hosts y routers. Dichas reglas permiten a los dispositivos reenviar los paquetes generados por los hosts a través de múltiples routers hasta llegar al host de destino correcto (Odom, 2014).

Puesto que IPv6 define direcciones con 128 bits, escribir estas en binario constituiría un problema por la dimensión de esta, razón por la cual se define un formato hexadecimal más corto que requiere como máximo 32 dígitos hexadecimales, y se aplican una serie de reglas para abreviar esta nomenclatura (Odom, 2014).

**Tabla 2.3** Notación para direcciones IPv6 en Hexadecimal

<b>Ejemplo direcciones IPv6</b>
2345:1111:2222:3333:4444:5555:6666:AAAA
2000:1:3:4:5:6:A
FE80::1

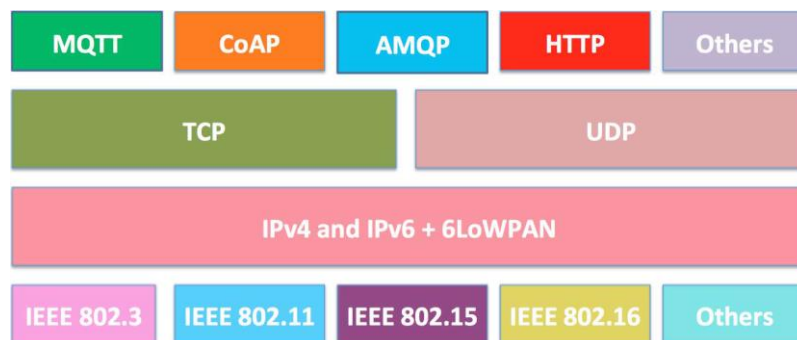
**Fuente:** (Odom, 2014)

Si bien es cierto, uno de los beneficios más importantes es el amplio rango de direcciones que ofrece, también existen otros aspectos importantes que mejoran el protocolo IPv6, tales como una administración mucho más fácil, mejor enrutamiento, cabecera más simple lo que deriva en un enrutamiento más eficiente, integración de autenticación, entre otros más (Salazar & Silvestre, 2016).

Hay que mencionar que IPv6 coexistirá a través de algún tiempo con IPv4, y su despliegue se realizará de forma gradual y ordenada, entonces los dispositivos de cliente, equipos de red, aplicaciones, contenidos y servicios deben adaptarse a esta transición y a su vez soportar la nueva versión del protocolo de internet IPv6 (Salazar & Silvestre, 2016).

#### 2.6.4 Tecnologías y protocolos de la capa de Aplicación

Uno de los requerimientos para el funcionamiento de sistemas IoT concierne al protocolo de comunicación y su latencia para garantizar de forma óptima su propósito, es así que en la presente sección se describen dos de los principales protocolos cuyo fin es el intercambio de datos y es posible enfocarlos para IoT, se refiere al protocolo HTTP y MQTT; el primero, diseñado para petición e intercambio de datos permitirá comprender la lógica aplicada a este tipo de protocolos, con lo cual se enfocará en MQTT como aquel protocolo empleado en la presente investigación para soluciones IoT.



**Figura 2.11** Stack de protocolos para IoT  
Fuente: (Naik, 2017)

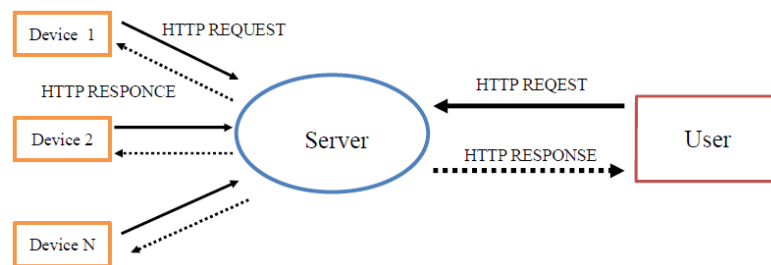
##### 2.6.4.1 HTTP

HTTP o protocolo de transferencia de Hipertexto constituye el protocolo de comunicaciones diseñado para los servicios de internet basado en el paradigma cliente servidor. Diferentes tipos de datos, imágenes, textos etc. Se envían a través de HTTP a través de internet todos los días. HTTP se basa en TCP, es decir garantiza que los datos transmitidos de un dispositivo a otro no se corrompan y la integridad de los datos está asegurada. HTTP es un protocolo abierto el cual puede ser leído por aplicaciones diseñadas para tal efecto y que generalmente se refiere a los navegadores. Como se

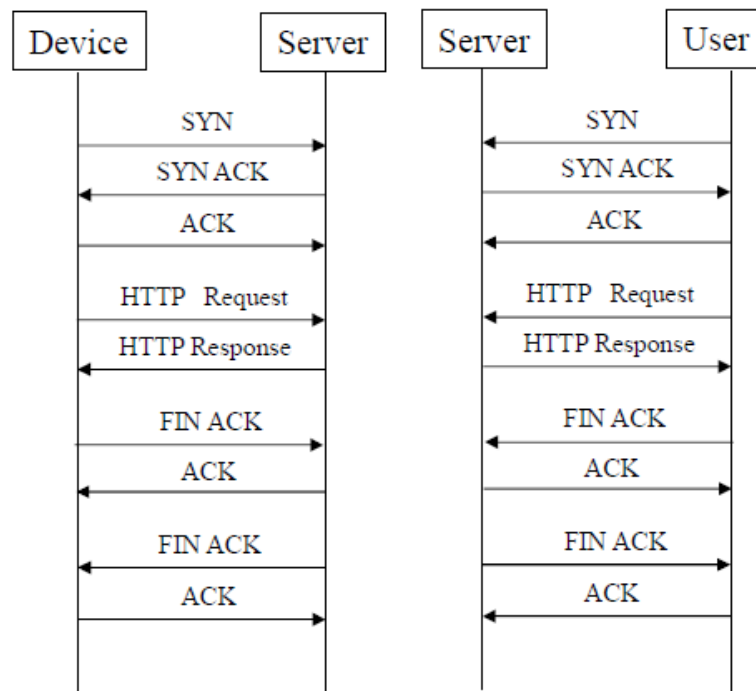


mencionó anteriormente, al basarse en el modelo cliente servidor se manejan operaciones de solicitud y respuesta. El comando solicitud es iniciado desde el cliente hacia el servidor, y el comando de respuesta es enviado desde el servidor al cliente, todo este intercambio de mensajes y datos forman bloques de datos con un formato específico conocido como mensaje HTTP (Naik, 2017).

Ahora bien, si se analiza HTTP para los escenarios requeridos en IoT, el protocolo tiene el cargo de transferir gran cantidad de pequeños paquetes, y es aquí donde la sobrecarga del protocolo HTTP ocasiona problemas tales como el gran consumo de recursos de red y también grandes retrasos.



**Figura 2.12** Esquema de sistema IoT usando HTTP  
**Fuente:** (Yokotani & Sasaki, 2016)

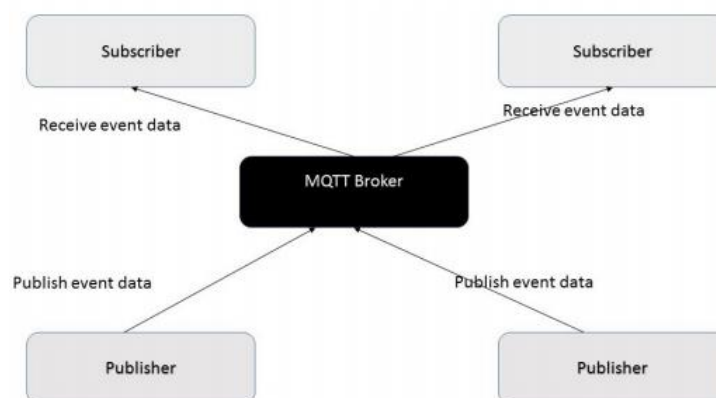


**Figura 2.13** Secuencia de comunicaciones en HTTP  
**Fuente:** (Yokotani & Sasaki, 2016)

De acuerdo con la Fig. 2.13 y según lo detalla (Yokotani & Sasaki, 2016), HTTP opera a través de TCP/IP por lo que proporciona una comunicación confiable, pero para las conexiones establecidas en TCP, se liberan en cada acceso debido a que los datos a los que se accede se transfieren en función de la dirección IP y la URL, y su relación de cambia dinámicamente, entonces, después de varios intercambios de mensajes para la liberación de una conexión esta es terminada. En resumen, la comunicación para IoT induce a una gran sobrecarga y consumo de recursos durante este periodo, por ello es necesario conocer otro protocolo que considere estos defectos y los solucione para su correcto funcionamiento en sistemas IoT.

#### 2.6.4.2 MQTT

MQTT (Message Queue Telemetry Transport), inventado y desarrollado por IBM en los años 90 es un protocolo de mensajería ligera de la capa aplicación que posee una sobrecarga muy liviana ya que cuenta con un tamaño de encabezado fijo de 2 bytes. El protocolo funciona bajo el principio de publicación – suscripción. MQTT es adecuado para IoT M2M (Máquina a Máquina) y redes sensores inalámbricos (Wukkadada, Wankhede, Nambiar, & Nair, 2018).



**Figura 2.14** Modelo Publicación - suscripción de MQTT.  
**Fuente:** (Dhall & Solanki, 2017)

El protocolo MQTT se ejecuta sobre TCP / IP, y como se mencionó anteriormente maneja un tamaño de paquete de datos con un mínimo de sobrecarga, esto hace que el consumo de la fuente de alimentación también sea bajo. Este protocolo se caracteriza porque

independientemente de los datos, los puede transmitir en varias formas, ya sea como datos binarios, texto, XML, o JSON, usando el modelo publicación / suscripción (Atmoko, Riantini, & Hasin, 2017).

Los elementos involucrados en el modelo son por una parte el servidor central denominado bróker y los suscriptores. Cuando los suscriptores inician una conexión TCP/IP con el bróker, esta se mantiene hasta que el cliente la finaliza, así mismo el servidor mantiene un registro de los suscriptores conectados. El modelo publicación / suscripción los dispositivos suscriptores también pueden obtener información publicada por un determinado dispositivo, lo que permite tener tipologías uno a uno o a su vez de uno a muchos.

Una de las entidades que maneja el protocolo MQTT es el denominado bróker, el cual recibe todos los mensajes de los clientes y a su vez los dirige hacia aquellos que es de interés los datos, en cambio el cliente puede ser un dispositivo de IoT o una aplicación que tiene la capacidad de procesamiento de este tipo de datos. En el modelo un cliente se puede suscribir a un “tema” de mensajería y si otro cliente publica mensajes sobre ese tema el bróker lo que hará es redirigir los mensajes de tema a estos clientes suscritos. En MQTT los mensajes están organizados en “temas”, estos permitirán que los clientes puedan acceder a únicamente determinados mensajes en función de su suscripción (Coria, 2020).

## **2.7 Computación en la nube y plataformas IoT**

El Cloud Computing se refiere a proporcionar computación, almacenamiento, aplicaciones y en general servicios TI a través de Internet, posiblemente residiendo de forma remota, de modo que los usuarios no deben tener en cuenta la ubicación física del servidor o el almacenamiento en función a sus necesidades (Sehgal & Bhatt, 2018).

De la misma manera (Sehgal & Bhatt, 2018) define los siguientes modelos de servicio para el Cloud Computing:

- **Software como servicio:** Este modelo se centra en los usuarios finales de la nube, se proporciona acceso a una determinada aplicación de modo que varios usuarios la pueden ejecutar, es necesario tener instalado un software en los dispositivos de usuario, en este modelo es el que proporciona menos control en la nube.
- **Plataforma como servicio:** está centrado en desarrolladores de aplicaciones y en las diferentes necesidades informáticas en base a las etapas del proyecto en desarrollo. Los servicios son cumplidos por servidores que pueden variar en sus características computacionales de acuerdo con requerimientos del usuario.
- **Infraestructura como servicio:** aquí se proporciona acceso directo a hardware virtualizado o en contenedores, es decir, servidores con especificaciones tales como el CPU, memoria y almacenamiento son puestos a disposición a través de la red.

La computación en la Nube proporciona considerables beneficios para diversas aplicaciones que requieren de cómputo y almacenamiento específico. En lo que tiene que ver con IoT aporta a los usuarios y aplicaciones formas fáciles de llevar a cabo el desarrollo de un determinado sistema. El Cloud Computing logra proporcionar de manera virtual toda una infraestructura que integra dispositivos, almacenamiento, herramientas de análisis, plataformas de visualización etc., y cuando es necesaria una expansión se la puede realizar sin mayores complicaciones incluso sin tener que comprar nuevo equipamiento.

## **CAPITULO 3 . Evaluación y desarrollo de plataforma IoT para el monitoreo de variables ambientales aplicadas al sector agropecuario.**

En el presente capítulo se desarrolla el trabajo de titulación, que en base al marco teórico presentado en el capítulo 2, se presentará una comparativa de los principales componentes de un sistema IoT, todo esto enfocado a su aplicabilidad en el sector agropecuario. De la misma manera se crea una plataforma que permita la lectura y visualización de variables ambientales, esta será implementada en un servicio en la nube, es decir permitirá recolectar datos enviados desde varios dispositivos, que para este caso se valdrá de hardware en una placa microchip IoT y también simulados a través de software que puede interpretar los protocolos usados en IoT. Todo esto no permitirá obtener un panorama del uso de este tipo de soluciones en la industria agropecuaria donde a los procesos se puede agregar un tipo de inteligencia que creará valor y permitirá optimizar los mismos.

### **3.1 Aplicaciones IoT en para el sector agropecuario.**

La agricultura forma parte de la civilización humana y sirve para proporcionar un medio de vida a los agricultores. El surgimiento de la agricultura inteligente impulsada por IoT permite el surgimiento de tareas, seguimiento y adquisición de datos ambientales de una gran variedad y volumen, seguido del paso e intercambio a través de la red, y todo esto permitirá tomar decisiones a corto y largo plazo basados en análisis de inteligencia artificial, así como también llevar a cabo acciones de forma remota con la ayuda de robots de campo. La aplicación de IoT a este sector también puede ayudar a la predicción de los rendimientos y con ello predecir los flujos de valores económicos, también la detección temprana de enfermedades para los cultivos y así tomar medidas preventivas oportunas (Kalla, Porambage, & Liyanage, 2019).

El objetivo de implementar IoT es la optimización máxima de la tierra cultivable y de sus insumos, en sí, mejorar e impulsar el desarrollo de la sociedad actual, y ser una solución viable al aumento de la demanda de alimentos debido el incremento de la población mundial. En la presente sección se centrará en describir algunas aplicaciones tanto para la denominada agricultura y ganadería inteligente, esperando crear un panorama claro del aporte que genera involucrar la internet de las cosas para optimizar la producción en estos sectores.

La implementación de sensores y dispositivos inteligentes permite recolectar información precisa del cultivo, suelo y variables climáticas en tiempo real, esto ayuda a optimizar la producción y calidad de los productos. También ayuda a la reducción de costos de producción, y toda la información recolectada para a ser almacenada para la elaboración de informes estadísticos que a su vez permitirán llevar a cabo acciones correctivas o preventivas (Mora & Urrego, 2018).

Entre varias de las aplicaciones de IoT para agricultura se tiene: agricultura de precisión, preparación y siembra inteligente de semillas, riego inteligente, información meteorológica, fertilización inteligente, cosecha inteligente, etc. Referente a la ganadería se tiene aplicaciones de mantenimiento inteligente del ganado, gestión inteligente que se ocupa del seguimiento de los animales, control inteligente de la salud, alimentación inteligente, gestión de forrajes. etc. Para desarrollar más estos temas se presentan a continuación una mejor descripción de varias aplicaciones consideradas de interés directo y que aportan al sector objeto de estudio, recalando que son a nivel general y no son las únicas que se pueden implementar.

### **3.1.1 Sistemas de gestión de Riego.**

En la agricultura moderna se requieren sistemas de riego inteligentes para optimizar el uso del agua en agricultura y actividades ligadas. Estos sistemas son integrados en tiempo real, usan datos de pronóstico del clima

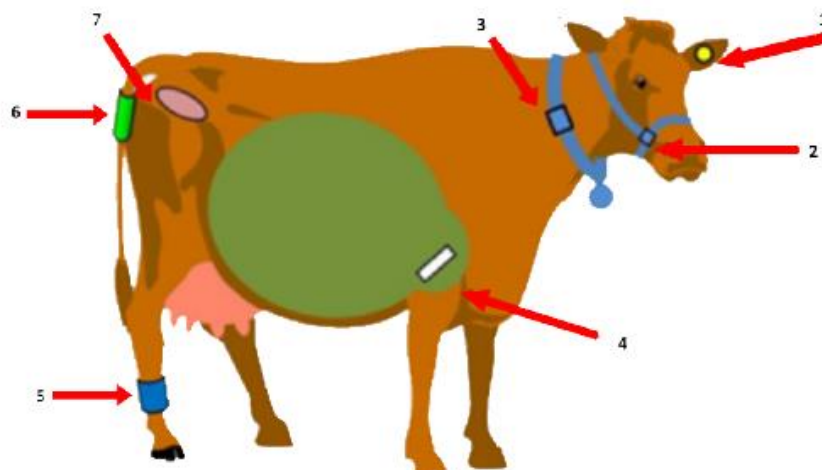
y le permiten al agricultor el control del sistema desde cualquier parte del mundo a través de la red en sincronización con sensores de humedad instalados en campo. Para el agricultor significa reducción de costos y conservación de recursos hídricos limitados (Ray, 2017)

### 3.1.2 Control de plagas y enfermedades

Según lo indica (Ray, 2017), el uso controlado de pesticidas y fertilizantes permite aumentar la calidad de los cultivos y a su vez minimizar costos. Para el control del uso de plaguicidas se debe monitorear la probabilidad de aparición de plagas en los cultivos, esta actividad requiere la recopilación de información de enfermedades y plagas de insectos con el uso de sensores para posteriormente realizar un procesamiento de la información obtenida e identificar la enfermedad respectiva, llevar a cabo las medidas necesarias, controlar a la plaga responsable y con esto el agricultor salve su inversión.

### 3.1.3 Monitoreo de ganado

IoT se puede usar para el monitoreo del movimiento de una manada de semovientes y con ello controlar como se produce el pastoreo en una determinada área, o hacer el seguimiento en tiempo real de las actividades de estos, alertando comportamientos e interacciones inusuales.



**Figura 3.1** Sensores más comunes para monitoreo de ganado: 1 Ubicación, 2 Comportamiento de alimentación, 3 Detección de celo, 4 Temperatura, PH, 5 Podómetro, 6,7 Detección de parto

**Fuente:** (Coria, 2020)

Para el sector ganadero también se tienen algunas aplicaciones interesantes, tales como el monitoreo del estado de salud de los animales, detección de celos, estado de partos entre otras muchas otras aplicaciones que pueden beneficiar al productor ganadero.

#### **3.1.4 Monitoreo de la calidad del agua**

El seguimiento de la calidad del agua es posible con IoT y con el despliegue de sensores que posean comunicación inalámbrica. El sistema mide físicamente parámetros químicos presentes en el agua tales como la temperatura, PH, turbidez, conductividad, oxígeno, etc. Estas variables son visualizadas a través de internet usando servicios en la nube (Ray, 2017).

#### **3.1.5 Monitoreo del suelo**

Conocer las propiedades que posee el suelo es crucial para desarrollar actividades agrícolas, y el monitoreo en tiempo real de la condición del suelo aporta una ventaja para los agricultores.

#### **3.1.6 Agricultura de precisión**

La denominada agricultura de precisión puede ser obtenida llevando a cabo el uso de tecnologías como UAV (Unmanned Aerial Vehicle) y Drone, aplicados a cultivos. Estos sistemas despliegan inteligencia artificial a través de sensores y cálculos de trayectorias de vuelo para la detección de condiciones climáticas en el aire. Además, permiten el monitoreo continuo de la zona de análisis y su propio estado mientras este en vuelo, de tal manera que se crean mapeos y soportes del sistema en tiempo real. Estas técnicas buscan lograr un óptimo rendimiento y control de cada terreno.





**Figura 3.2** Agricultura de precisión  
**Fuente:** (MundoGEO, 2021)

Son varios los aspectos positivos que se generan producto de la agricultura de precisión, así por ejemplo se tiene: la maximización del rendimiento por cada zona, la minimización de usos de pesticidas y fertilizantes, reducción de usos de recursos hídricos y energéticos, sistemas de medición de riego entre muchos otros (Joint Research Centre (JRC) of the European Commis, 2014).

### 3.2 Comparativa de tecnologías de comunicación para IoT

En el capítulo 2 se presentó características para algunas tecnologías de comunicación de interés para la presente investigación, la presente sección se remite a realizar una comparativa de las mismas para tener juicio de valor en la elección de una tecnología para una determinada aplicación IoT, que para este caso está enfocada al sector agropecuario.

**Tabla 3.1** Comparación de algunas tecnologías de comunicación para IoT

Tecnología	WIFI	Bluetooth	LoRa	ZigBee
<b>Banda de Frecuencia</b>	2.4 / 5 GHz	2.4 GHz	868/900 MHz	868/915 MHz /2.4 GHz
<b>Data Rate</b>	1 Mbps – 6.75 Gbps	1-24 Mbps	0.3-50 Kbps	40-250 Kbps
<b>Rango de transmisión</b>	20-100 m	8-10 m	< 30 Km	10-100 m

<b>Consumo de energía</b>	Alto	Bajo	Muy Bajo	Bajo
<b>Costo</b>	Alto	Bajo	Alto	Bajo

**Fuente:** Elaboración propia basada en (Ray, 2017)

La tabla anterior resume y permite hacer una comparativa de las principales características que defienden a una determinada tecnología. Así mismo, tanto para agricultura como ganadería se puede tener un ejemplo de aplicación y considerar los requisitos básicos para una aplicación IoT.

Si se toma en cuenta que las áreas en donde se despliegan procesos tanto de agricultura como ganadería, estas son generalmente de gran amplitud, siendo así LoRa y específicamente LoRaWan la tecnología que constituye una alternativa bastante idónea para la implementación de IoT en estos sectores. Obviamente cada caso de análisis es independiente y debe plantearse la solución siempre considerando todos los aspectos, tales como costos, velocidades, rangos de cobertura etc.

### **3.3 Comparativa de servicios en la nube**

La implementación de un sistema IoT requiere una plataforma en la nube que ofrezca algunos servicios que permita un adecuado desarrollo del sistema. A nivel global gran cantidad de proveedores, de estos son tres los que abarcan un gran dominio de los servicios en la nube y para IoT, estos son: Amazon Web Services, Microsoft Azure, y Google Cloud. Como usuarios, es necesario seleccionar de forma concienzuda los servicios ofrecidos por las empresas, con ello escoger las más adecuada a estos intereses, demandas y objetivos. Es por ello por lo que a continuación se presenta una tabla que permite comparar algunas de las características ofrecidas por las plataformas antes mencionadas, cuyo dominio en el mercado es imperioso.

**Tabla 3.2** Comparación de algunas de las plataformas de servicios en la Nube

<b>Tecnología</b>	<b>Amazon Web Services (AWS)</b>	<b>Microsoft Azure</b>	<b>Google Cloud Platform</b>
<b>Tipo de servicio en la nube</b>	IAAS SAAS PAAS, (presenta mejores contribuciones con IAAS)	IAAS SAAS PAAS, presenta mejores contribuciones con PAAS)	IAAS SAAS PAAS, presenta mejores contribuciones con PAAS)
<b>Internet de las cosas</b>	IoT Core IoT Analytics IoT 1 click IoT Button IoT Device Defender IoT Device	IoT Edge, IoT Hub Time series Insight Stream analytics	IoT Core
<b>Tecnología de Virtualización</b>	XEN Virtualization Technology	Hyper-V Hypervisor Virtualization technology	KVM Hypervisor Virtualization technology
<b>Interfaz de usuario</b>	Basada en Web	Basada en Web, Application Program Interface(API), Consola	Basada en Web, Application Program Interface(API), Consola
<b>Tipos de Máquinas Virtuales</b>	1. Propósito general 2. Computación Optimizada. 4. Memoria Optimizada. 4. Almacenamiento Optimizado. 5. Computación Acelerada	1. Propósito general. 2. Computación Optimizada. 3. Memoria Optimizada. 4. Almacenamiento Optimizado. 5. Computación de alto rendimiento.	1. Máquina estándar 2. Máquinas de alta memoria. 3. Máquina de alta CPU. 4. Máquinas de mega memoria.
<b>Costo</b>	On-Demand, facturación por segundo	Pago por uso.	Pago por uso, facturación por segundo bajo demanda.

**Fuente:** elaboración propia basada en: (Opara, 2019) (Wankhede, Talati, & Chinchamatpure, 2020)

La tabla anterior presenta una breve comparación de tres servicios en la nube reconocidos a nivel mundial como líderes en este campo, cada

proveedor posee múltiples servicios que dependiendo del proyecto y de costos el usuario podrá inclinarse por alguno de ellos.

Los tres grandes tecnológicos de servicios en la nube brindan servicios para IoT y aún no existe un claro predominante que se imponga en esta competencia para el mercado, pero además existen múltiples empresas que buscan marcar un camino y ser partícipe de un sector de mercado.

A continuación, se indica el funcionamiento de una plataforma IoT, desde su organización hasta la generación de código para un sistema embebido destinado a la lectura de sensores y su transmisión al servidor MQTT.

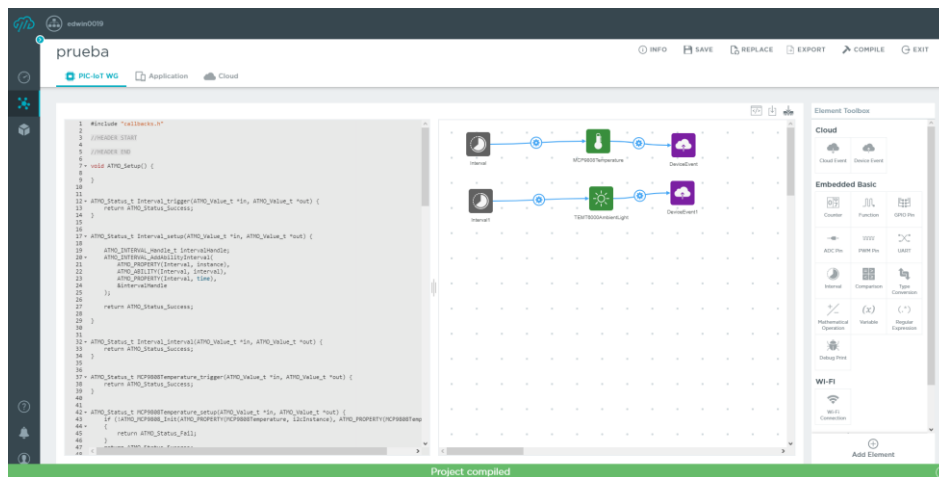
### **3.4 Plataforma IoT Atmosphere**

Atmosphere es una plataforma de servicios IoT basado en la nube que ayuda a la creación de aplicaciones IoT de forma rápida e intuitiva a los usuarios, además permite manejar los aspectos del desarrollo, es decir desde el firmware de los dispositivos embebidos hasta la visualización de los datos en la nube.

El desarrollo de una aplicación IoT basado en la plataforma Atmosphere se indica a continuación, se usa la placa de desarrollo Wifi de Microchip basada en el MCU PIC24FJ128GA705 misma que cuenta con el módulo WINC1510 Wi-Fi que permite su conexión con la red, además cuenta con dos sensores integrados para medir la temperatura ambiente y nivel de iluminación ambiental. En todo caso la plataforma cuenta ya con las librerías necesarias para el desarrollo y programación de la mencionada placa.

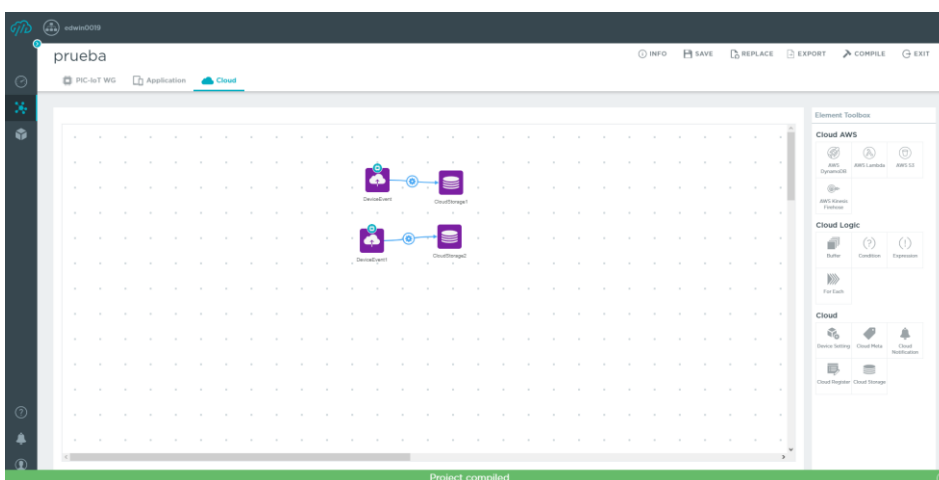
Para el desarrollo del firmware del sistema embebido, se deben establecer los intervalos para leer los datos de los sensores tanto de temperatura ambiental como de luz en la placa. Los intervalos de lectura de estos valores se los define en las propiedades de cada bloque, para este caso se definen en 500ms. Indicados los sensores que se van a usar en la placa y

su evento para envío de datos hacia la nube, se procede a realizar la compilación del proyecto, y la plataforma generará el código para la aplicación, y más aún permitirá descargar el archivo .hex que servirá para la programación del microcontrolador. La siguiente figura muestra el ambiente de desarrollo y el código generado para la programación del microcontrolador.



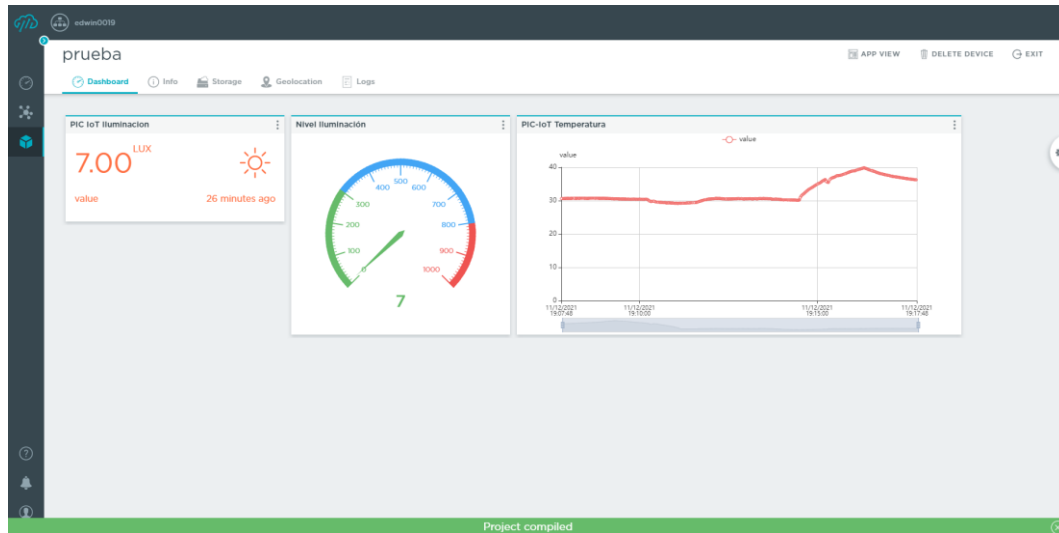
**Figura 3.3** Ambiente de desarrollo de Atmosphere para sistemas embebidos  
**Fuente:** Elaboración propia

Así mismo, Atmosphere cuenta con una plataforma de desarrollo para establecer conexiones hacia la nube. La figura 3.4 indica el ambiente de desarrollo para lectura y almacenamiento de datos provenientes de los dispositivos, en este caso los dos sensores de la placa de desarrollo.



**Figura 3.4** ambiente de desarrollo para conectividad en la nube Atmosphere  
**Fuente:** Elaboración propia

Ya en operación de la aplicación, Atmosphere muestra el dashboard con la información que está recibiendo de los sensores tal como se lo indica en la figura 3.5.



**Figura 3.5** Dashboard IoT en la plataforma Atmosphere  
**Fuente:** Elaboración propia

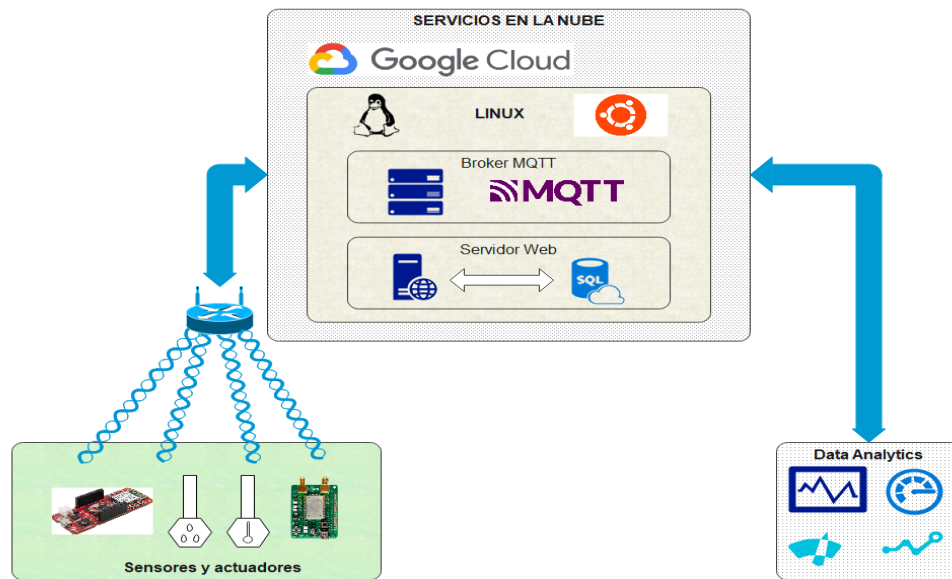
Conocer como es la operación de la plataforma Atmosphere da la pauta para establecer el funcionamiento general y como deben desarrollarse las aplicaciones destinadas a resolver problemáticas del internet de las cosas.

A continuación, se indaga y se describe una arquitectura general para llevar a cabo el desarrollo general de aplicaciones IoT basadas en la nube y más aun, el proceso para llevar a cabo las principales configuraciones y desarrollos para obtener un sistema funcional que involucre todo lo tratado hasta aquí.

### **3.5 Arquitectura IoT general para aplicaciones.**

Tomando como base el desarrollo y funcionamiento general de la plataforma IoT indicada anteriormente, en la presente sección se presenta una arquitectura de IoT de propósito general en base a la investigación desarrollada y que puede servir a nivel general para la implementación de aplicaciones IoT, pero para estos fines está orientada a aplicaciones del

sector agropecuario. La arquitectura propuesta está basada en el modelo de 4 capas descrito en el capítulo II, es decir, se recopilan los datos necesarios y se transmiten a través de la red hacia un back-end implementado en la nube donde los mismos son procesados, y que a continuación se pueden visualizar en una plataforma front-end accesible para el usuario final.



**Figura 3.6** Topología de arquitectura IoT  
Fuente: Elaboración propia

De la topología presentada, primeramente, se observan los dispositivos o equipos encargados de la toma de datos correspondientes a variables ambientales u otras de interés, estos dispositivos también pueden ser actuadores que ejecuten o inicien una determinada tarea que se requiera y se configure en un sistema. Igualmente, para llevar a cabo el desarrollo de la investigación se continuará con el uso de la placa Microchip PIC-IoT cuyo funcionamiento se indicó igualmente con la plataforma descrita en la sección anterior, se hará uso de los sensores de temperatura y luz, además se usará su diseño capaz de establecer conexión con la nube. Para la transmisión de datos se arma una trama JSON que contiene el valor de las variables que se lean de los sensores y son transmitidas hacia la nube para su visualización. Existe software que es capaz de funcionar con el protocolo MQTT y enviar información en dicho formato, con lo cual se puede

simular un dispositivo IoT, dicha simulación también se considera dentro de la presente investigación y permitirá evaluar la arquitectura de una plataforma destinada a IoT.

La sección de comunicaciones y tal como se lo ha descrito anteriormente estará regida por las diversas tecnologías existentes en la actualidad, y como sea la necesidad de una aplicación IoT se optará por una determinada tecnología para su aplicación, en el caso de la placa PIC-IoT de Microchip, esta tiene incorporada una tarjeta WIFI que permitirá tener conexión a internet.

Lo que se refiere a la implementación del servidor Broker MQTT, servidor web, base de datos, en definitiva a lo que generalmente se conoce como Back-End y Front-End será desplegado a través de la plataforma de servicios en la nube que en este para este caso se usa Google Cloud, desde la cual se despliega una máquina que corre el sistema operativo Linux - Ubuntu, aquí se instalarán e implementarán los servicios necesarios para el correcto funcionamiento de estas aplicaciones, así mismo se implementa una base de datos basada en SQL, misma que tiene como objetivo el almacenar los datos que envían los sensores, información de estos, claves de acceso, usuarios etc., esta base de datos podrá ser consultada o modificada tanto desde el Back-End o el Front-End.

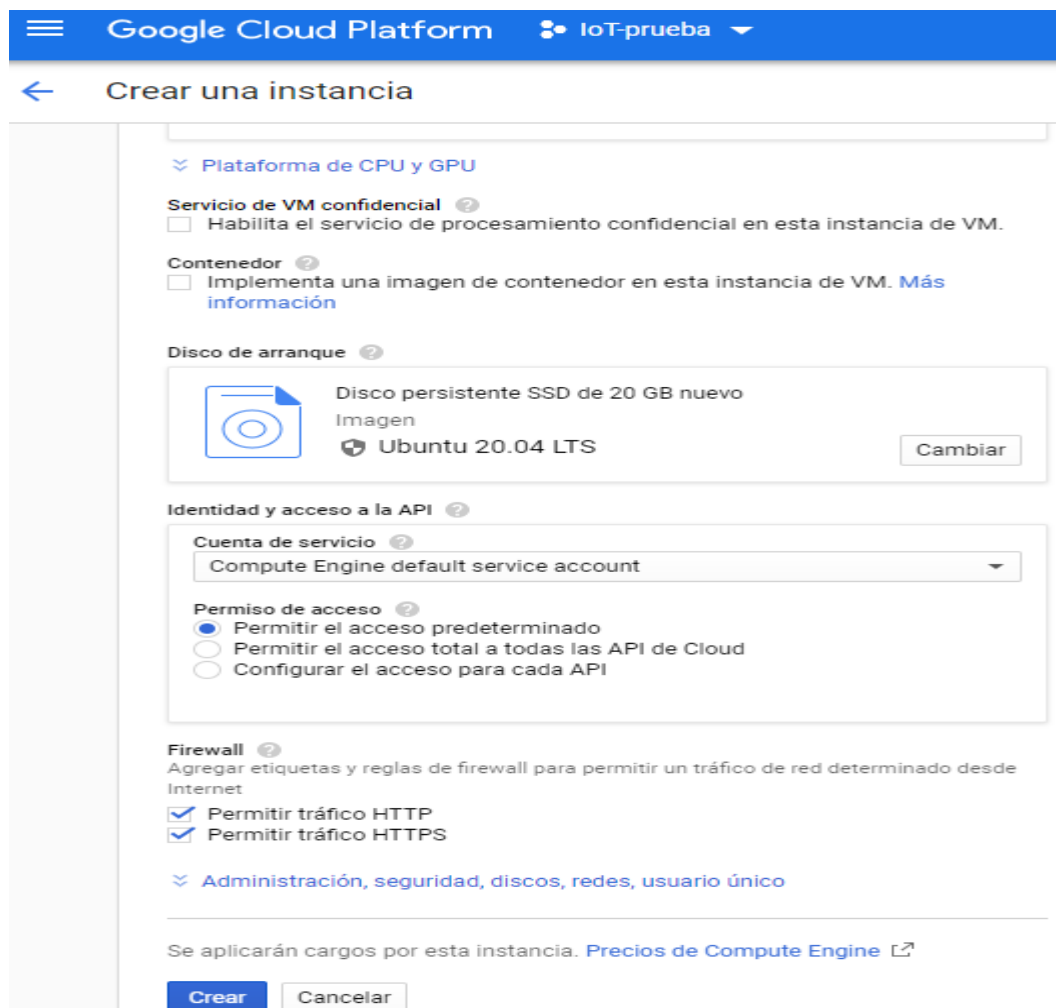
### **3.6 Implementación de servidor**

Los servicios basados en Cloud permiten virtualizar hosting dedicado y para estas necesidades se tiene la opción de configurar el hardware de acuerdo con tales necesidades, además se pueden agregar servicios especializados en cloud tales como servicios de bases de datos, machine learning etc.

En Google Cloud se crean instancias de máquinas virtuales, la figura 3.7 indica la creación de la instancia de la máquina en la que se carga la imagen de Ubuntu 20.04 LTS, se le dota de un disco SSD de 20 GB, además se



habilita el tráfico tanto HTTP como el HTTPS, que serán indispensables para llevar a cabo el proyecto.



**Figura 3.7** Creación de instancia en Google Cloud

**Fuente:** Elaboración propia

Una vez creada la instancia con su sistema operativo, es importante el tener en cuenta la denominada como IP Externa, esta dirección es a la cual se debe apuntar para la conexión del servidor y sus servicios, así mismo la plataforma permitirá tener una conexión SSH para poder iniciar a trabajar en la instalación de los paquetes y librerías necesarios. La dirección IP externa que Google Cloud asigna también es necesario que se configure como fija, cuyo propósito es que si en un determinado momento que la instancia es reiniciada se pueda tener acceso con la misma dirección IP asignada.

Estado	Nombre ↑	Zona	Recomendaciones	En uso por	IP interna	IP externa	Conectar
<input checked="" type="checkbox"/>	<a href="#">instance-1</a>	us-central1-a			10.128.0.2 (nic0)	<a href="#">34.66.125.25</a>	SSH

**Figura 3.8** Información de instancia creada en Google Cloud.  
Fuente: Elaboración propia

Establecida la conexión con el servidor a través de SSH se deben realizar las actualizaciones respectivas para sistemas Linux para inmediatamente realizar la instalación del servidor WEB, para este caso se ha optado por NGINX, mismo que es open source, de alto rendimiento y ofrece los contenidos de sitios web de manera rápida, ha esto se suma una fácil configuración. Este servidor web puede gestionar miles de conexiones simultáneas a una gran velocidad y escalabilidad.

Se instala el servidor ejecutando el siguiente comando:

```
edwin019q@instance-1:~$ sudo apt install nginx
```

Otro de los servidores que se instala en el servidor de mensajes IoT MQTT, para este caso se hace uso del servidor EMQX también de código abierto, este hará de Broker para manejar las múltiples conexiones de los dispositivos que se suscriban o publiquen al mismo. Como se lo describió anteriormente, un Broker a su vez en un intermediario que facilita la comunicación de los dispositivos con algún otro tipo de suscriptor. EMQX presenta varias características que brindan comodidades a los desarrolladoras, entre ellas se puede indicar que se respalda en una plataforma comunitaria open source, es decir que su funcionamiento está apoyado por toda una comunidad, además tiene gran soporte para una gran cantidad de dispositivos sin disminuir su rendimiento, trabaja con conexiones seguras y a su vez con protocolos de seguridad robustos,

cuenta con un dashboard intuitivo de fácil manejo en donde se pueden monitorear los estados de las conexiones y también del mismo Broker. También permite la integración con bases de datos entre muchas otras características que aportan al desarrollo efectivo de comunicaciones para la internet de las cosas.

Desde la página oficial de EMQX se descarga el paquete de instalación y posteriormente proceder a su instalación e inicialización tal como se indica a continuación:

```
edwin019q@instance-1:~$ sudo dpkg -i emqx-ubuntu20.04-4.2.7-x86_64.deb
Selecting previously unselected package emqx.
(Reading database ... 64363 files and directories currently installed.)
Preparing to unpack emqx-ubuntu20.04-4.2.7-x86_64.deb ...
Unpacking emqx (4.2.7) ...
Setting up emqx (4.2.7) ...
Adding group `emqx' (GID 120) ...
Done.
Adding system user `emqx' (UID 113) ...
Adding new user `emqx' (UID 113) with group `emqx' ...
Not creating home directory `/var/lib/emqx'.
Processing triggers for systemd (245.4-4ubuntu3.11) ...

edwin019q@instance-1:~$ sudo emqx start
EMQ X Broker 4.2.7 is started successfully!
edwin019q@instance-1:~$
```

**Figura 3.9** Instalación e inicialización de servidor MQTT  
Fuente: Elaboración propia

Puesto que una plataforma diseñada para IoT se basada en tecnologías de BackEnd y FrontEnd, y el desarrollo de estas se lo maneja a través del lenguaje JavaScript, entonces para el lado del FrontEnd la interpretación del lenguaje lo realiza el navegador web, pero para el caso del BackEnd es necesario que la interpretación la realice otra aplicación para este caso será NODE.JS, entorno de programación que permitirá ejecutar JavaScript del lado del servidor y así facilitar la creación de aplicaciones escalables. En conclusión, se instala NODE.JS sobre Linux y a través de un solo lenguaje (JavaScript) se desarrollará lo que es tanto el FrontEnd y BackEnd.

NODE.JS sobre Linux constituye un framework de JavaScript que aporta al desarrollo de plataformas en la parte del BackEnd, dentro del presente análisis de los sistemas IoT, y específicamente con el protocolo MQTT este

cumple su papel de suscriptor, es decir puede obtener información de las publicaciones de los dispositivos, pero este también puede cumplir la función de publicador, que sería el caso inverso cuando desde una plataforma se desee enviar datos a los dispositivos para ejecutar una determinada tarea. NODE permite ejecutar JavaScript desde el servidor.

NPM es conocido el manejador de paquetes de node.js, además permite instalar las librerías necesarias dentro del entorno de desarrollo, la instalación de Node.js se realiza con los comandos que se indican a continuación:

```
edwin019q@instance-1:~$ sudo apt install npm
```

Para comprobar que Node.js se encuentra correctamente instalado se ejecutan los siguientes comandos.

```
edwin019q@instance-1:~$ node -v
v10.19.0
edwin019q@instance-1:~$ npm -v
6.14.4
```

Instalado NODE.JS sobre un sistema Linux, se procede a su configuración y de esta manera a través de un solo lenguaje de programación se podrá realizar el desarrollo de FrontEnd y BackEnd.

La máquina(instancia) creada en la nube contendrá los directorios donde albergaran los ficheros para el correcto funcionamiento de los servicios de BackEnd y FrontEnd, dichos servicios para ambas tecnologías deben ser capaces mantener una comunicación entre sí y para ello se establecen servicios que poseen la capacidad de establecer comunicación entre el BackEnd y el FrontEnd, esta tecnología es conocida como WebSocket.

```
edwin019q@instance-1:~/backend$ npm i socket.io
```

### 3.7 ExpressJS

Es una framework para Nodejs, mismo que sirve y ayuda a crear aplicaciones web gracias varias funcionalidades, es una herramienta que ayuda a generar las APIS, aquí en donde se aplica la lógica de la programación, para estos propósitos, que en estos casos va desde la comunicación hasta la interacción con la base de datos. En definitiva, este framework ayuda a tener una estructura base y con ello facilitar la gestión de peticiones http, de sesiones, cookies, etc. Y en este caso también se usa para el desarrollo de API's de diferentes propósitos.

En el servidor ExpressJS se instala con el siguiente comando:

```
npm i express
```

Express hace uso del modelo-vista-controlador, arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en componentes diferentes. En este flujo de trabajo el modelo es desarrollado para poder manipular, realizar gestión y actualizaciones de los datos, este tiene una conjunción de operación con la denominada vista, en la cual se muestran al usuario final la ventanas, pantallas o formularios que se creen por motivo de una solicitud, finalmente lo que se refiere al controlador es el encargado de gestionar la comunicación entre el modelo y la vista, ósea las instrucciones que se generen son recibidas, atendidas y procesadas.

No únicamente se habla de peticiones HTTP, también se usan los denominados websokets, tecnologías que permiten la creación de un canal bidireccional sobre un socket o un hilo, esto es usado para tener la comunicación entre los navegadores y los servidores web.

El archivo index se busca por defecto cuando no se especifica el archivo exacto de inicio, en este caso existe la necesidad de trabajar con varios

archivos de desarrollo por lo tanto se tiene que analizar y especificar todas las rutas que se creen.

Cuando se requieren procesos en tiempo real como es el caso para IoT, los sensores tienen la capacidad de enviar información al broker a través de MQTT con ya se lo ha descrito anteriormente, el broker envía información a NODE y su vez este realiza un cambio de protocolo, es decir, los datos que se reciben a través de MQTT, a través de Express se cambian por medio de una petición por medio de Web Socket hacia el FrontEnd.

### **3.8 Bases de datos**

Unas de las partes fundamentales que conforman una plataforma IoT es lo que tiene que ver con el correcto procesamiento, estructuración y almacenamiento de la información, es decir lo que tiene que ver con una base de datos. Dentro de un sistema IoT los datos generados por los dispositivos de demás características de la plataforma son procesados, relacionados, estructurados y almacenados a fin de que puedan ser consultados de acuerdo con las características selectivas que se indique en base los requerimientos. Otro de los aspectos a considerar y que debe permitir el acceso a la base de datos implementada es el control de usuarios y dispositivos, es decir, se deben implementar funciones que permitan la creación, modificación y acceso a grupos específicos de usuarios o dispositivos, quienes serán aquellos con privilegios de acceso a los recursos que proporcione la plataforma. El tener la capacidad de consultar los usuarios y dispositivos que están autorizados para el uso de la plataforma IoT crea una primera base de seguridad, en donde el apoyo de una base de datos sólida es fundamental.

La plataforma de base de datos que se usa es MySQL, que funciona con tablas relacionales y que cabe destacar de este sistema que es open source, lo que permite que su utilización sea gratuita y se pueda manipular con libertad. Además, se usa bases de datos relacionales ya que poseen

gran estabilidad, gran soporte y presentan un relativa sencillez para su instalación en los sistemas operativos que para este caso es Linux.

Se instala la base de datos en la máquina en la nube de la siguiente manera:

```
edwin019q@instance-1:~$ sudo apt install mysql-server
```

Para realizar la configuración de opciones de seguridad recomendadas se ejecuta el siguiente comando:

```
edwin019q@instance-1:~$ sudo mysql_secure_installation
```

Después de realizar todas las configuraciones se procede a crear la base de datos:

```
mysql> CREATE DATABASE iot CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
```

Se verifica y se cambia la base de datos en uso.

```
mysql> show DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| iot |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.01 sec)

mysql> USE iot;
Database changed
```

Creaciones de tablas

```
mysql> CREATE TABLE usuarios (
-> id INT AUTO_INCREMENT NOT NULL PRIMARY KEY,
-> nombre VARCHAR(100) NOT NULL,
-> correo_electronico VARCHAR(60) NOT NULL,
-> contrasena VARCHAR(100) NOT NULL,
-> activo TINYINT NOT NULL DEFAULT 1,
-> fecha_hora_creacion DATETIME NOT NULL
-> );
Query OK, 0 rows affected (0.09 sec)

mysql> █
```

```
mysql> CREATE TABLE dispositivos (
-> id INT AUTO_INCREMENT NOT NULL PRIMARY KEY,
-> nombre VARCHAR(50) NOT NULL,
-> token VARCHAR(150) NULL,
-> fecha_hora_creacion DATETIME NOT NULL,
-> usuario_id INT NOT NULL,
-> FOREIGN KEY (usuario_id) REFERENCES usuarios (id)
-> );
Query OK, 0 rows affected (0.08 sec)
```

La creación de tablas y registros en la base de datos se lo hará a través de programación. ORM constituye un modelo de programación que permite realizar un mapeo de las estructuras de una base de datos relacional y con ello facilitar que el sistema sea interactivo con el usuario. En programación las tablas se vuelven objetos, con lo cual ya es posible la manipulación de datos, en este caso se instala el ORM denominado *sequelize*.

```
edwin019q@instance-1:~$ sudo npm i -g sequelize-cli
```

```
edwin019q@instance-1:~$ npm i sequelize
```

```
edwin019q@instance-1:~$ sequelize init
```

Con el ORM instalado se puede mapear los campos de la base de datos

### 3.9 JWT (JSON web token)

Basado en JSON, es un estándar abierto que permite crear un token cuyo propósito es enviar datos entre aplicaciones o servicios y a su vez garantizar que sean válidos y seguros.

El usar JWT proporciona seguridad al envío de datos, ya que la información se cifra haciendo uso del algoritmo HS256



### 3.10 VueJS

Del lado del FrontEnd existen tecnologías que permiten el desarrollo de lo que el usuario podrá visualizar e interactuar, una de las conocidas y que se analiza es VueJS, framework de JavaScript para crear interfaces de usuario. Se trata de una capa añadida a JavaScript formada por herramientas convencionales de trabajo y un lenguaje particular que permite crear aplicaciones de forma rápida, sencilla y práctica. Dispone de tres elementos que lo hacen característico, el Template, característica que permite tener un archivo VUE, con un extracto de html, uno de CSS y un extracto de JavaScript, que juntas conforman las tecnologías para el desarrollo Web, establecidas en un solo archivo, lo que a su vez se traduce como simplicidad en los desarrollos. También presenta la característica de “Reactividad”, indispensable para que los eventos desencadenen en tiempo real la visualización de este, es decir evitar la recarga de la página para poder visualizar los cambios. VueJS se encarga de modificar únicamente el elemento al que se hizo el cambio, de esta manera no es necesario renderizar toda la página sino más bien los elementos que presentan los cambios, con ello permite el desarrollo de páginas de forma más efectiva. Bajo VueJS se elaboran los desarrollos de páginas web, y este se monta sobre el servidor web, mismo que se encarga de la resolución de la página web y mostrar las gráficas en tiempo real, en este caso se instala a NGINX como servidor web.

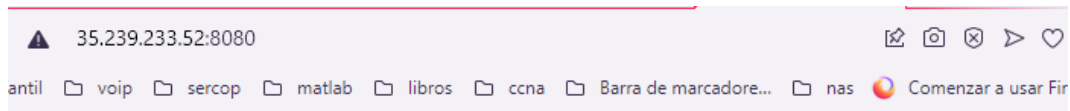
Se instala VUE a través de la ejecución de los siguientes comandos:

```
edwin019q@instance-1:~$ sudo npm i -g @vue/cli
```

```
edwin019q@instance-1:~$ vue create frontend
```

```
Vue CLI v4.5.13  
? Please pick a preset: Default ([Vue 2] babel, eslint)
```

Se ingresa al navegador web y con la dirección del servidor y el puerto 8080 se verifica el correcto funcionamiento de Vue.js.



## Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project, check out the [vue-cli documentation](#).

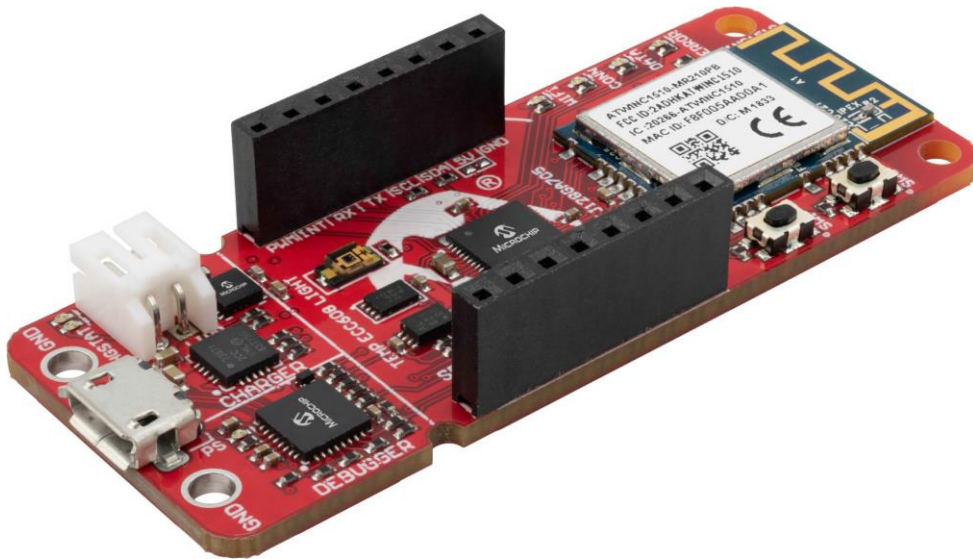
**Figura 3.10** Interfaz de Inicio Vue.js  
Fuente: Elaboración propia

### 3.11 Dispositivos IoT

Ya se han presentado los componentes que crean un sistema de internet de las cosas, ahora se centra en detallar y profundizar lo que respecta a los dispositivos encargados de la lectura de variables del medio ambiente y otras. Un nodo IoT debe estar conformado por tres bloques, primero la lógica o inteligencia, segundo un sistema de seguridad que proporcione confiabilidad en los datos que interactuará y por último la conexión, es decir, características que en conjunto conforman un dispositivo provisto de inteligencia capaz de transmitir datos hacia el exterior garantizando su seguridad.

En el presente trabajo investigativo se usa primeramente la placa PIC-IoT provista por Microchip para el desarrollo de dispositivos IoT, que a su vez cumple con todas las características descritas anteriormente, es decir presenta un bloque de lógica o inteligencia proporcionado por el microcontrolador PIC24FJ128GA705, también cuenta con el chip ATECC608 que tiene como propósito hacer un cifrado de los datos que maneja la placa, y en cuanto a la capa de comunicaciones esta placa esta

provista por la tarjeta WIC1510 que permite tener conexión WIFI y poder acceder a las redes de datos.

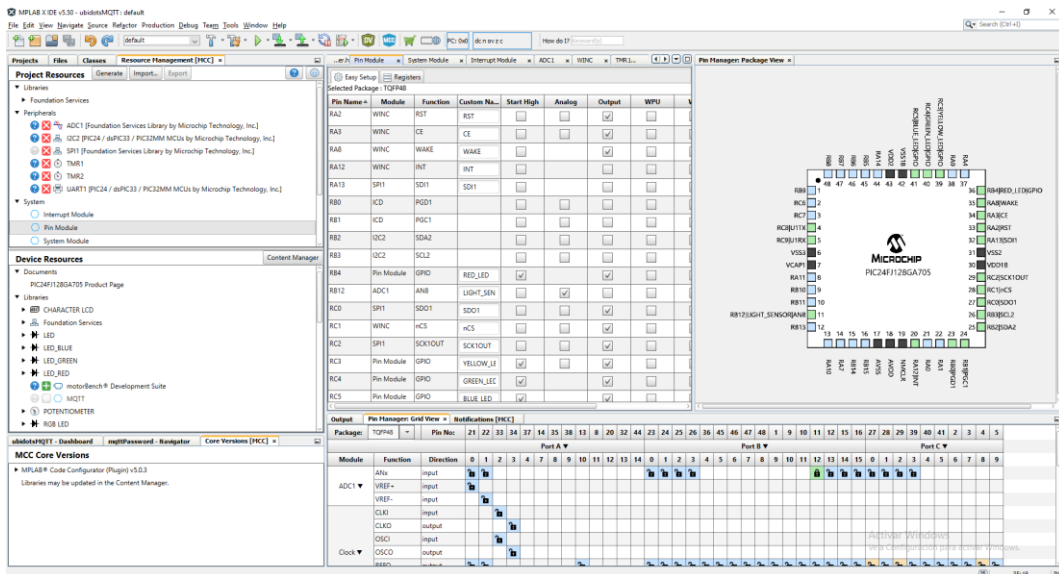


**Figura 3.11** Placa PIC-IoT WG Development Board  
**Fuente:** (Incorporated, Microchip Technology, 2019)

Para el desarrollo de sistemas IoT, la placa es provista de dos sensores, uno mide la temperatura ambiente y otro el nivel de iluminación a través de un sensor de luz incorporado. La placa cuenta con un programador USB integrado, que permite realizar la programación del microcontrolador al estilo drag and drop, y entre otras muchas características que la hacen ideal para demostrar el funcionamiento de un sistema IoT.

### 3.12 Programación de placa IoT

Microchip ofrece un amplio conjunto de herramientas y aplicativos destinados a facilitar el desarrollo de soluciones basadas en microcontroladores, una de ellas es MCC (MPLAB Configurator), entorno de programación gráfica que genera código C transparente fácil de analizar y a su vez insertar en el proyecto.



**Figura 3.12 Entorno MCC**  
Fuente: Elaboración propia

En la figura anterior se indica el entorno que ofrece el aplicativo MCC para el proceso de desarrollo en sistemas embebidos, se tiene acceso a múltiples características propias del microcontrolador tal como los timers, sistemas de comunicación serial, ADC, librerías IoT, etc. De tal manera que cuando se establezcan todos los criterios de desarrollo MCC genera un nuevo firmware para la placa, donde el código generado es distribuido en varios ficheros ya que es considerable la cantidad de codificación para el funcionamiento de una aplicación, esto es comparable con la forma de desarrollo que se presentó anteriormente en la plataforma Atmosphere, donde a través de bloques se estableció el funcionamiento de la aplicación. A continuación, se analiza aquellas partes importantes que constituyen pilares fundamentales para la correcta ejecución de los objetivos previstos para la placa.

El siguiente código muestra la función de envío de datos en una trama JSON hacia la nube y en sí a la plataforma IoT, esta a su vez llama a las funciones de lectura tanto de temperatura como de nivel de luz para posteriormente hacer la publicación de dichas lecturas en la trama JSON al bróker MQTT. Con esta función se obtiene los datos de los sensores

```

void sendToCloud()
{
    static uint8_t json[70];
    float temperature = sensor_readTemperature();
    //float temperature = 23.1;
    uint16_t light = sensor_getLightValue();

    int len = sprintf(json, "{\"temperatura\":%1.2f, \"luz\":%d}", temperature, light);
    if(len > 0)
    {
        MQTT_CLIENT_publish((uint8_t*)json, len);
    }
}

```

En cuanto a la lectura de los sensores, tanto de temperatura como de luz, se establece una comunicación serial con el microcontrolador para el envío de datos que son procedentes de los sensores y los conversores análogos a digital provistos en mismo microcontrolador.

```

float sensor_readTemperature()
{
    uint16_t temp;
    float tempF;
    I2C2_MESSAGE_STATUS status = I2C2_MESSAGE_PENDING;

    I2C2_MasterWrite(&REG_TA, 1, MCP9809_ADDR, &status);
    I2C2_MasterRead(&word, 2, MCP9809_ADDR, &status);
    while(status == I2C2_MESSAGE_PENDING);

    if(status == I2C2_MESSAGE_COMPLETE)
    {
        temp = _htons(word);
        temp = temp & 0x0FFF;
        tempF = (float) temp / 16.0;
        printf("\n\rTemperature: %1.2f\n\r", tempF);
        return tempF;
    }
    return 0;
}

uint16_t sensor_getLightValue()
{
    return ADC1_GetConversion(LIGHT_SENSOR);
}

```

Otro aspecto importante que merece su análisis en el código es la comunicación entre el cliente MQTT y el bróker, se genera un archivo el cual contiene características para integrar la información de este dispositivo con la plataforma en la nube, en ella se crea un token que permite el acceso

en relación con el dispositivo. Otro de los parámetros importantes a considerar es el denominado tópicos que es exclusivo para cada uno de los dispositivos que establecen conexión en a la plataforma.

```
#include "bsd_mqtt_adapter.h"

#define MQTT_CID_LENGTH      100
#define MQTT_TOPIC_LENGTH   38
#define MQTT_UBIDTOS_TOKEN_LENGTH 36
#define MQTT_SUBSCRIBE_TOPIC_LEN 38

SOCKET      tcpClientSocket;
bool        sendSubscribe = true;
uint8_t     recvBuffer[50];
uint32_t    mqttUbidotsIP;
uint8_t     cid[10]; //client ID

uint8_t     mqttUsername[MQTT_UBIDTOS_TOKEN_LENGTH] = "admin";
uint8_t     mqttPassword[200] = "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJkZXZpY2VfaWQiOiJlZmMjIOMDk3Nzh9.h4j816";
uint8_t     mqttTopic[MQTT_TOPIC_LENGTH] = "/dispositivos/0Gywi0L6c0";
uint8_t     mqttSubscribeTopic[MQTT_SUBSCRIBE_TOPIC_LEN] = "/v1.6/devices/pic-iot-wg/led/lv";
uint8_t     counter = 0;
```

En lo referente con la conexión de la tarjeta inalámbrica, desde MCC se indica el SSI y el password de la red WIFI a la cual debe conectarse la tarjeta. En parte del código indicado a continuación se indican funciones que cuyos argumentos solicitan los valores en mención, estos a su vez vienen contenidos dentro de las cabeceras del código C generado. Resuelto todo esto la placa tendrá conexión a red y salida a la Internet, por ende, comunicación con una plataforma IoT.

```
bool wifi_connect()
{
    uint8_t ssid[20];
    uint8_t pass[20];
    uint8_t authType[2];
    int8_t ret;

    strcpy(ssid, CFG_MAIN_WLAN_SSID);
    strcpy(pass, CFG_MAIN_WLAN_PSK);
    sprintf(authType, "%d", CFG_MAIN_WLAN_AUTH);

    ret = m2m_wifi_connect((char*)ssid, sizeof(ssid), atoi(authType), (char*)pass, M2M_WIFI_CH_ALL);
    if(ret != M2M_SUCCESS)
    {
        printf("ERROR: Failed to Init Wi-Fi Connection\n\r");
        return false;
    }
    return true;
}
```

El microcontrolador debe establecer una conexión TCP hacia el servidor MQTT, y en la función que a continuación se indica se establecen tanto el puerto de comunicación como la dirección IP a donde apunta el dispositivo, en este caso, la dirección IP del servidor que se encuentra en la nube.

Como se puede visualizar en el código, la dirección IP a la cual se establece la conexión efectivamente es la dirección del servidor, pero con la particularidad que esta expresado en notación hexadecimal, y a su vez el puerto para la comunicación es 1883 del bróker MQTT. Establecida la conexión, la comunicación se efectúa a través de los denominados web sockets obteniendo ya una aplicación del Internet de las cosas.

```
void connectMQTTsocket()
{
    struct sockaddr_in strAddr;
    int8_t ret;

    strAddr.sin_family      = AF_INET; //IPv4
    strAddr.sin_port        = _htons(1883);
    strAddr.sin_addr.s_addr = _htonl(0x22427D19); //34.66.125.25  S

    mqttContext *context = MQTT_GetClientConnectionInfo();
    socketState_t socketState = BSD_GetSocketState(*context->tcpClientSocket);

    printf("Creating Socket...\n");

    if(socketState == NOT_A_SOCKET)
    {
        *context->tcpClientSocket = socket(AF_INET, SOCK_STREAM, 0);
        if(*context->tcpClientSocket >= 0)
        {
            packetReceptionHandler_t* sockInfo = getSocketInfo(*context->tcpClientSocket);
            if(sockInfo != NULL)
            {
                sockInfo->socketState = SOCKET_CLOSED;
                printf("Socket Created\n");
            }
        }
    }
}
```

El código generado a través de MCC contiene muchas más funcionalidades cuyo análisis no es objeto del presente trabajo, y lo indicado anteriormente permitirá obtener una visión adecuada de lo que la placa IoT debe llevar a cabo para el correcto funcionamiento del sistema. Después de compilar el código y generar el archivo hex se procede a la programación del microcontrolador; que al final cada cierto periodo de tiempo envía datos a la nube provenientes de las lecturas de temperatura y de luz a través de los sensores que posee implementados.





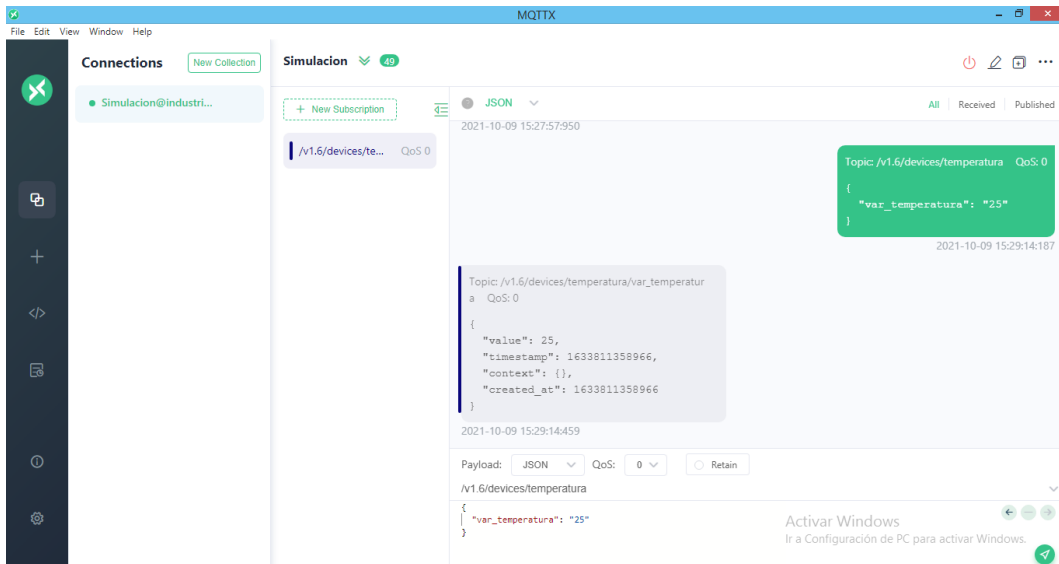
**Figura 3.13** Placa Pic-IoT en operación  
Fuente: Elaboración propia

Todo lo desarrollado e indicado anteriormente se muestra con el propósito de obtener un sistema IoT dentro de un ambiente de prueba sencillo pero que a su vez brinda las pautas de que características debe poseer un sistema diseñado para la operación en un entorno industrial.

### 3.13 Simuladores de clientes MQTT

En la actualidad existe también software especializado para simular lo que la placa (hardware) lleva a cabo. El funcionamiento rige bajo el mismo principio, es decir se establece la comunicación con el Broker MQTT a través de la dirección IP y del número de puerto usado para la comunicación, se establece la contraseña y el tópico de registro de dispositivo y una vez que se dispone de la conexión se envían las tramas JSON con la información de las variables que se visualizaran en la plataforma. Esta también es una forma válida de obtener un ambiente de pruebas para el desarrollo de soluciones a través de IoT. Entre algunos de los aplicativos existente están MQX, MQTT-fx, etc., incluso es válido hacer el desarrollo de un aplicativo propio que cumpla tal función, cuyo desarrollo puede realizárselo en Python, Matlab, Labview etc.





**Figura 3.14** Entorno de comunicación EMQx  
Fuente: Elaboración propia

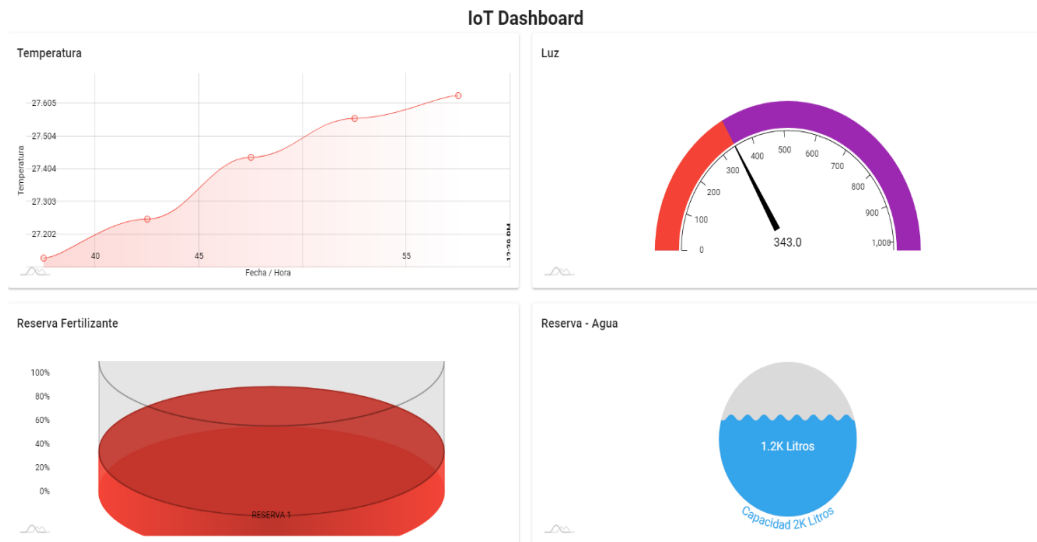
## **CAPITULO 4 . Plataforma IoT para el monitoreo de variables en el sector agropecuario.**

Uno de los propósitos de una plataforma para el internet de las cosas es mostrar adecuadamente el monitoreo de datos provenientes de sensores o dispositivos que hagan referencia a determinadas variables en un determinado sistema, que como se lo ha planteado en la presente investigación mejoraría los procesos y toma de decisiones en los sectores dedicados a la agricultura y ganadería.

La figura 4.1 muestra un dashboard desarrollado para el monitoreo y visualización de variables de interés en una determinada aplicación , en el mismo se muestran lecturas que el usuario final puede conocer inmediatamente, que en este caso se indican la temperatura ambiental, nivel de iluminación, nivel de depósitos de fertilizantes y nivel de agua en reserva; en todo caso, son solo algunas de múltiples variables que pueden monitorizarse e indicarse en tiempo real, con lo cual dependerá del usuario final el uso y objetivos que se plantee asignar a la información obtenida.

Tener claro el objetivo que se busca al implementar un sistema de internet de las cosas marca el inicio para el desarrollo de las funcionalidades que debe dotársele al sistema. Tal como se ha indicado, existen plataformas ya constituidas que en sus servicios brindan varias herramientas a la hora de implementar una determinada aplicación, esto reduce el tiempo de implementación, claro está, a cambio del pago de un valor mensual, pero que a su vez no exime de las restricciones que los propietarios de la plataforma declaren. Otra opción que se tiene es la de llevar a cabo todo el desarrollo de la aplicación bajo propios medios, pero que a su vez demanda de mayores y múltiples conocimientos especializados, por ende debe estar a cargo una persona con conocimientos sólidos y adecuados del tema; también en cierto que considerablemente demanda más tiempo de implementación, pero brinda algunas ventajas como el poder adaptarse a la infraestructura de redes en la nube o de propia de la empresa en la cual se implemente, además en el desarrollo propio se permite gestionar en

base de requerimientos propios de cada empresa, así, la Fig. 4.1 muestra una tablero de variables para una situación de monitoreo de variables que pueden ser de utilidad en la agricultura y ganadería.



**Figura 4.1** Resultado Dashboard IoT

Fuente: Elaboración propia

Las gráficas del tablero de la figura anterior son basadas en la librería de *amcharts4* ([www.amcharts.com](http://www.amcharts.com), 2021) en la cual el desarrollador dispone de varias graficas cuya codificación se basa en JavaScript, y se pueden adecuar a un proyecto en desarrollo.

Un punto que se podría considerar como desfavorable para una empresa al momento de hacer uso de esta segunda opción es el mantenimiento que una plataforma requiere, esto implica la adición al equipo de trabajo de la institución de una o varias personas con conocimientos adecuados del tema que a su vez que se encarguen de la operación y mantenimiento del sistema.

## CONCLUSIONES

En un sistema IoT convergen varias tecnologías que permiten la lectura y monitoreo de variables de interés para la toma de decisiones en los sectores productivos. Para el sector agropecuario brinda la oportunidad de tener un control eficiente del desarrollo de producción, es decir, se puede implementar un sistema que ayude a llevar y conocer eficazmente los procesos y desarrollo de actividades para el sector en mención. Es posible tener desde un sencillo tablero de control con pocas variables hasta grandes sistemas en operación con inteligencia artificial para automatizar procesos. En cumplimiento del objetivo general de la presente investigación, también se detalla la arquitectura general que debe poseer una plataforma orientada a estos servicios para un correcto funcionamiento, y de la evaluación de esta arquitectura se lleva a cabo la metodología para realizar la implementación de un sistema IoT que en tiempo real permita al usuario disponer de datos fiables y llevar a cabo toma de decisiones.

En referencia al primer objetivo específico, se ha desarrollado un marco teórico de la situación actual de los sistemas de internet de las cosas, aquí se indican desde topologías sugeridas, tecnologías de comunicaciones, plataformas para el desarrollo de soluciones etc, para que, estudiados estos fundamentos teóricos estar en la capacidad de entender la arquitectura que gobierna un sistema IoT para su implementación.

En cumplimiento al segundo objetivo específico, se indican diferentes topologías para IoT, así mismo, se detallan las tecnologías involucradas en el proceso de desarrollo de un sistema para IoT, además de otros componentes adicionales requeridos, y es que, en base a ello se propone una arquitectura de propósitos generales para ser aplicada a la construcción de un sistema de internet de las cosas, que para el enfoque dado a la investigación se plantea su uso en aplicaciones propias del sector agrícola y ganadero.

Finalmente, en el desarrollo del presente trabajo se realiza la implementación de un sistema IoT en un ambiente de pruebas, pero que a su vez permite entender el desarrollo de un sistema de mayor magnitud. Se indican los tableros de gráficas de variables en plataformas IoT, tanto en plataformas pago como es el caso de Atmosphere. También indica el proceso de llevar a cabo el desarrollo de un sistema propio a través de tecnologías web, cuya infraestructura puede estar en la nube o dentro de una red empresarial. Para llevar a cabo esto debe existir personal con conocimientos suficientes de tecnologías web tanto a nivel de BackEnd y FrontEnd, así como también del conocimiento de protocolos como MQTT para la comunicación de los dispositivos finales con el servidor. También empresas como Microchip han desarrollado tarjetas que permiten llevar a cabo ambientes de desarrollo. La placa PIC-IoT de microchip brinda múltiples prestaciones para evaluar una aplicación en concreto para IoT, con ella se pudo recolectar información de variables de luz y de temperatura, para ser enviada y visualizada en una plataforma en la nube haciendo uso del protocolo MQTT y conexión WIFI.

## RECOMENDACIONES

Se recomienda tener un análisis claro de la tecnología que se hará uso en la capa de comunicaciones, su elección determinada a una tecnología que marcará el correcto desempeño de un sistema IoT. Para ello es necesario realizar un análisis correcto y adecuado de la zona en la cual los dispositivos o sensores se instalarán. Del análisis siempre se optará por una determinada tecnología que sobresaldrá sobre otra.

También se recomienda un análisis adecuado a la elección de la infraestructura sobre la que se implementará la plataforma IoT. Como ya se ha visto puede montarse sobre plataformas en la nube con las múltiples ventajas que involucra, o ser montadas sobre infraestructura propia de una empresa. En definitiva, hay que hacer un balance que vislumbre el cumplimiento de los propósitos y objetivos planteados a la hora de hacer la elección de una aplicación para IoT.

Se recomienda hacer uso de clientes MQTT simulados por software, esto ayuda a tener de forma rápida una evaluación del funcionamiento de un sistema IoT en desarrollo, lo que permite identificar y hacer correcciones a tiempo de manera oportuna.

Al tener un campo bastante amplio de aplicabilidad para IoT se recomienda prestar suficiente atención a procesos primordiales y que requieren de optimización para su aplicación en la industria.

También se debe hacer un análisis y evaluación de la eficiencia energética de los dispositivos destinados a la adquisición de información de variables en campo, así, por ejemplo, en el sector agropecuario se monitorea grandes campos de producción, por lo que es necesario que el consumo energético sea óptimo.

## Bibliografía

- Aliero, M., Ahmad, A., Kalgo, U., & Aliero, S. (2020). An Overview of Internet of Things: Understanding the Issues and Challenges of a More Connected World. *International Journal of Computing and Communication Networks*, 2, 1-11.
- Arroyo, A. (2019). *Estudio Comparativo de Plataformas Middleware Open-Source de Internet de las cosas (IoT) Basado en Métricas Cuantitativas y Cualitativas*. Universidad Andina Néstor Cáceres Velásquez. Obtenido de <http://repositorio.uancv.edu.pe/handle/UANCV/2264>
- Atmoko, R., Riantini, R., & Hasin, M. (2017). IoT real time data acquisition using MQTT protocol. *Journal of Physics: Conference Series*, 853, 012003. doi:10.1088/1742-6596/853/1/012003
- Bandyopadhyay, D., & Sen, J. (2011). Internet of Things: Applications and Challenges in Technology and Standardization. *Wireless Personal Communications*, 58, 49-69. doi:<https://doi.org/10.1007/s11277-011-0288-5>
- Barriga, Á. (2017). *Oportunidades de negocio en el ámbito del internet de las cosas*. Obtenido de <http://hdl.handle.net/10662/6569>
- Camacho, L., Rey, C., Araujo, G., Chavez, D., Cordova, C., & y varios. (2008). *Redes Inalámbricas para Zonas Rurales* (1 ed.). Lima, Peru: Pontificia Universidad Católica del Perú.
- Coria, A. (2020). *Aplicaciones de IoT para el mundo real* (Primera ed.). Buenos Aires: mcelectronics.
- Curran, K. (2009). *Understanding the Internet. A Glimpse Into the Building Blocks, Applications, Security and Hidden Secrets of the Web* (1 ed.). Chandos Publishing.
- Dhall, R., & Solanki, V. (2017). An IoT Based Predictive Connected Car Maintenance Approach. *International Journal of Interactive Multimedia and Artificial Intelligence*, 4, 16-22. doi:10.9781/ijimai.2017.433

- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*(7), 1645–1660. doi:10.1016/j.future.2013.01.010
- Hemjal, M. (2019). *Sigfox Based Internet of Things: Technology, Measurements and Development*. Incorporated, Microchip Technology. (2019). *PIC-IoT WG Development Board User Guide*. Obtenido de <https://ww1.microchip.com/downloads/en/DeviceDoc/PIC-IoT-WG-User-Guide-50002856A.pdf>: <https://ww1.microchip.com/downloads/en/DeviceDoc/PIC-IoT-WG-User-Guide-50002856A.pdf>
- Jabraeil, M., Bahrami, B., Heidari, A., Allahverdizadeh, P., & Norouzi, F. (2020). IoT Architecture. En *Towards the Internet of Things: Architectures, Security, and Applications* (págs. 9-31). Springer International Publishing. doi:DOI: 10.1007/978-3-030-18468-1\_2
- Joint Research Centre (JRC) of the European Commis. (2014). PRECISION AGRICULTURE: AN OPPORTUNITY FOR EU FARMERS - POTENTIAL SUPPORT WITH THE CAP 2014-2020. *AGRICULTURE AND RURAL DEVELOPMENT*, 56.
- Joseph, O. (2018). *ULTRA-NARROWBAND INTERNET-OF-THINGS TECHNOLOGIES*. Tampere University of Technology. Obtenido de <https://trepo.tuni.fi/bitstream/handle/123456789/26178/Joseph.pdf?sequence=4&isAllowed=y>
- Kalla, A., Porambage, P., & Liyanage, M. (2019). Introduction to IoT. doi:DOI: 10.1002/9781119527978.ch1
- Khutsoane, O., Isong, B., & Abu-Mahfouz, A. (2017). IoT devices and applications based on LoRa/LoRaWAN. *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, 6107-6112. doi:10.1109/IECON.2017.8217061
- Kulkarni, S., & Kulkarni, S. (2017). International Journal For Science Technology And Engineering. En *Communication Models in Internet of Things: A Survey* (Vol. 3, págs. 87-91).



- Li, S., Xu, L., & Zhao, S. (2015). The internet of things: a survey. *Information Systems Frontiers*, 17, 243-259. doi:10.1007/s10796-014-9492-7
- Mora, M., & Urrego, K. (2018). *Monografía Internet de las cosas: Modelos de Comunicación, desafíos y aplicaciones*. Villavicencio, Colombia: Universidad de los Llanos.
- MundoGEO. (18 de 03 de 2021). *El mercado de la agricultura de precisión alcanzará los 3.700 millones de euros en todo el mundo en 2025*. Obtenido de mundogeo.com: <https://mundogeo.com/es/2021/03/12/el-mercado-de-la-agricultura-de-precision-alcanzara-los-3-700-millones-de-euros-en-todo-el-mundo-en-2025/>
- Naik, N. (2017). Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. *2017 IEEE International Systems Engineering Symposium (ISSE)*, (págs. 1-7). doi:10.1109/SysEng.2017.8088251
- Naveen, S. (2016). *Study of IoT: Understanding IoT Architecture, Applications, Issues and Challenges*.
- Odom, W. (2014). *CISCO CCENT/CCNA ICND1 100-101*. Madrid: Pearson Education S.A.
- Opara, C. (2019). *Cloud computing in Amazon Web Services, Microsoft Windows Azure, Google App Engine and IBM cloud platforms: A comparative study*. Near East University.
- Pino, S., Aguilar, H., Apolo, A., & Sisalema, L. (2018). Aporte del sector agropecuario a la economía del Ecuador. Análisis crítico de su evolución en el período de dolarización. Años 2000–2016. *Revista Espacios*, 39.
- Postel, J. (1981). Internet protocol.
- Ramírez, I., & Mazón, B. (2018). *Análisis de datos agropecuarios*. Machala: Machala: Universidad Técnica de Machala.
- Ramya, C., Shanmugaraj, M., & Prabakaran, R. (2011). Study on ZigBee technology. *2011 3rd International Conference on Electronics Computer Technology*. 6, págs. 297-301. IEEE.

- Ray, P. (2017). Internet of things for smart agriculture: Technologies, practices and future direction. *Journal of Ambient Intelligence and Smart Environments*, 9, 395-420. doi:10.3233/AIS-170440
- RedHat. (2018). *¿Qué es el middleware?* Obtenido de <https://www.redhat.com/es/topics/middleware/what-is-middleware>
- Salazar, J., & Silvestre, S. (2016). Internet de las cosas. *Techpedia. České vysoké učení technické v Praze Fakulta elektrotechnická. Scientific American*. (5 de Noviembre de 2007). Obtenido de How does Bluetooth work?: <https://www.scientificamerican.com/article/experts-how-does-bluetooth-work/>
- Sehgal, N., & Bhatt, P. (2018). *Cloud computing*. Springer.
- Song, S., & Issac, B. (2014-11). Analysis of WiFi and WiMAX and Wireless Network Coexistence. *International Journal of Computer Networks and Communications (IJCNC)*, 6. doi:10.5121/ijcnc.2014.6605
- Sosa, C. M., Tello, E., & Lara, D. (s.f.). Enfoque para Generar Aplicaciones Orientadas a Servicios para IoT mediante el Desarrollo Dirigido por Modelos. 10.
- Tschofenig, H., Arkko, J., Thaler, D., & McPherson, D. (2015). *Architectural Considerations in Smart Object Networking* (Vol. RFC 7452). doi:10.17487/RFC7452
- UIT-T. (06 de 2012). *UIT-T Rec. Y.2060 (06/2012) Descripción general de Internet de los objetos*. Obtenido de <http://handle.itu.int/11.1002/1000/11559>
- Vite, H., Vargas, O., Vargas, L., & Vargas, J. (2018). *INTERNET DE LAS COSAS APLICADO A LA PRODUCCIÓN AGROPECUARIA*. Guayaquil, Ecuador: Grupo Compás.
- Wankhede, P., Talati, M., & Chinchamatpure, R. (2020). Comparative study of cloud platforms-Microsoft Azure, Google Cloud Platform and Amazon EC2. *Journal of Research in Engineering and Applied Sciences*.
- Wukkadada, B., Wankhede, K., Nambiar, R., & Nair, A. (2018). Comparison with HTTP and MQTT In Internet of Things (IoT). *2018 International*

- Conference on Inventive Research in Computing Applications (ICIRCA)*, 249-253. doi:10.1109/ICIRCA.2018.8597401  
[www.amcharts.com](http://www.amcharts.com). (12 de 12 de 2021). Obtenido de [www.amcharts.com](http://www.amcharts.com):  
<https://www.amcharts.com>
- Yokotani, T., & Sasaki, Y. (2016). Comparison with HTTP and MQTT on required network resources for IoT. *2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, (págs. 1-6). doi:10.1109/ICCEREC.2016.7814989
- Zavaleta, S. (2020). *Diseño de una solución basada en el internet de las cosas (IoT) empleando Lorawan para el monitoreo de cultivos agrícolas en Perú*. Lima: Universidad Tecnológica del Perú. Obtenido de <https://hdl.handle.net/20.500.12867/2946>
- Zhang, Y.-c., & Yu, J. (Enero de 2013). A Study on the Fire IOT Development Strategy. *Procedia Engineering*, 52, 314-319. doi:10.1016/j.proeng.2013.02.146
- Zourmand, A., Kun Hing, A., Wai Hung, C., & AbdulRehman, M. (2019). Internet of Things (IoT) using LoRa technology. *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, 324-330. doi:10.1109/I2CACIS.2019.8825008

## **GLOSARIO DE TÉRMINOS**

**API:** Application Programming Interface

**AWS:** Amazon Web Services

**CDMA:** Code Division Multiple Access

**CSMA/CA :** Carrier Sense Multiple Access with Collision Avoidance

**CSMA/CD:** Carrier Sense Multiple Access with Collision Detection

**FSK:** Frequency Shift Keying

**GSM:** Global System For Mobile Comunication

**HTTP:** Hypertext Transfer Protocol

**IAAS:** Infraestructura como servicio

**IoT:** Internet of Things

**IP:** Internet Protocol

**ISM:** Industrial, Scientific and Medical

**JSON:** JavaScript Object Notation

**LAN:** Local Area Network

**LPWA:** Low Power Wide Area

**LTE:** Long Term Evolution

**MQTT:** Message Queue Telemetry Transport

**PAAS:** Plataforma como servicio

**RFID:** Radio Frequency Identification

**SAAS:** Software como servicio

**SIG:** Special Interest Group

**SOA:** Service-Oriented Architectures

**TCP:** Transmission Control Protocol

**TIC:** Tecnologías de la información y comunicación

**UAV:** Unmanned Aerial Vehicle

**URL:** Uniform Resource Locator

**UIT:** Unión Internacional de Telecomunicaciones

**WAN:** Wide Area Network

**XML:** Extensible Markup Language

## DECLARACIÓN Y AUTORIZACIÓN

Yo, **Quichimbo Angamarca Edwin Fernando**, con C.C: # **1104289457** autor del trabajo de titulación, **Evaluación de sistemas IoT para monitoreo de variables ambientales y su aplicación en la producción agropecuaria**, previo a la obtención del título de **Magíster en Telecomunicaciones** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, a los 11 días del mes marzo del año 2022



**Quichimbo Angamarca Edwin Fernando**

**C.C: 1104289457**



**REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA**

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

<b>TÍTULO Y SUBTÍTULO:</b>	Evaluación de sistemas IoT para monitoreo de variables ambientales y su aplicación en la producción agropecuaria.	
<b>AUTOR(ES)</b>	Quichimbo Angamarca Edwin Fernando	
<b>REVISOR(ES)/TUTOR</b>	MSc. Luis Córdova Rivadeneira; MSc. Edgar Quezada Calle / MSc. Manuel Romero Paz	
<b>INSTITUCIÓN:</b>	Universidad Católica Santiago de Guayaquil	
<b>FACULTAD:</b>	Sistema de Posgrado	
<b>PROGRAMA:</b>	Maestría en Telecomunicaciones	
<b>TÍTULO OBTENIDO:</b>	Magister en Telecomunicaciones	
<b>FECHA DE PUBLICACIÓN:</b>	11 de marzo de 2022	No. DE PÁGINAS: 92
<b>ÁREAS TEMÁTICAS:</b>	Internet de las cosas, Arquitecturas, Redes, Modelos, Protocolos, Tecnologías.	
<b>PALABRAS CLAVES/ KEYWORDS:</b>	IoT, MQTT, Google Cloud, Plataforma Atmosphere, HTTP, JSON, FronEnd, BackEnd, VUE.	
<b>RESUMEN/ABSTRACT:</b>	<p>Desde la medición y monitoreo de variables físicas son de interés para el sector agropecuario, su importancia radica en la gestión de procesos de producción y obtención de indicadores que muestran las fortalezas o debilidades en dichos procesos. En el presente proyecto se realiza un estudio y evaluación a nivel general de la arquitectura para sistemas IoT. Se enfoca en la investigación de soluciones de aplicación en el sector agrícola y ganadero. Se llevará a cabo la implementación de un dashboard en la plataforma IoT Atmosphere y en un servidor de infraestructura en la nube en este caso en Google Cloud en el cual se puedan visualizar mediciones de variables en tiempo real. Las pruebas se realizan a través de la placa de desarrollo de PIC-IoT de Microchip, y también a través de software que simula a clientes MQTT, como es el caso de MQTXX. Se procede con el envío de datos hacia plataformas las cuales muestran en tiempo real las lecturas de las variables monitorizadas. En el desarrollo del presente trabajo se evalúa lo que concierne a los protocolos y tecnologías que conforman una arquitectura funcional para sistema IoT. También se indican el proceso de implementación de determinados servicios en una etapa previa para el desarrollo de una plataforma IoT.</p>	
<b>ADJUNTO PDF:</b>	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO
<b>CONTACTO CON AUTOR/ES:</b>	<b>Teléfono:</b> +593-991793903	<b>E-mail:</b> edwin019q@gmail.com
<b>CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):</b>	<b>Nombre:</b> Romero Paz Manuel de Jesús	
	<b>Teléfono:</b> +593-994606932	
	<b>E-mail:</b> manuel.romero@cu.ucsg.edu.ec	
<b>SECCIÓN PARA USO DE BIBLIOTECA</b>		
<b>Nº. DE REGISTRO (en base a datos):</b>		
<b>Nº. DE CLASIFICACIÓN:</b>		
<b>DIRECCIÓN URL (tesis en la web):</b>		