

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA,
ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



**“DISEÑO DE UNA RED FOG BASADO EN INTERNET DE LAS COSAS
PARA MONITOREAR LA CONTAMINACIÓN EN LA BAHÍA DEL
LAGO TITICACA”**

TESIS

PRESENTADO POR:

EDWARD MIJAEL TTACCA HUALLA

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ELECTRÓNICO

PUNO – PERÚ

2017

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y
SISTEMAS

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA

“DISEÑO DE UNA RED FOG BASADO EN INTERNET DE LAS COSAS
PARA MONITOREAR LA CONTAMINACIÓN EN LA BAHÍA DEL
LAGO TITICACA”

TESIS PRESENTADA POR:

EDWARD MIJAEL TTACCA HUALLA

PARA OPTAR EL TITULO PROFESIONAL DE:
INGENIERO ELECTRÓNICO

FECHA DE SUSTENTACIÓN: 19-12-2017



APROBADA POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE : _____
Dr. JOSE EMMANUEL CRUZ DE LA CRUZ

PRIMER MIEMBRO : _____
M.Sc. GAVINO JOSÉ FLORES CHIPANA

SEGUNDO MIEMBRO : _____
Ing. JASMANY BUELAS CHAMBI

DIRECTOR / ASESOR : _____
Ing. FERDINAND EDGARDO PINEDA ANCCO

PUNO – PERÚ

2017

Área : Telecomunicaciones.

Tema : Diseño de Redes de última generación.

DEDICATORIA

Con aprecio a mis padres Valentin B. Ttacca, Feliciano Hualla a mis hermanos Rosa, Ismael, Yanet, Paul, Will y Dima por su apoyo que hicieron posible mi formación profesional y la culminación del presente trabajo. Así mismo, a mis docentes amigos y compañeros.

AGRADECIMIENTO

Mi agradecimiento a la Universidad Nacional del Altiplano, a la carrera Profesional de Ingeniería Electrónica por brindarme una formación de primer nivel, a mi Director de Tesis Ing. Ferdinand Edgardo Pineda Ancco por su apoyo en el desarrollo de este trabajo, a los Docentes que me brindaron sus conocimientos en los años de formación, a mis amigos Wily Reyes, Rudy Zapana, Abrahan Laura, Ronal Mamani y en especial a Ursula Mamani por su apoyo incondicional y a todos los que colaboraron de forma directa o indirecta con este trabajo.

ÍNDICE GENERAL

RESUMEN	15
ABSTRACT	16
CAPITULO I	17
INTRODUCCIÓN	17
1.1 Planteamiento de problema.....	18
1.1.1 Formulacion del problema.....	18
1.2 Justificacion del problema	19
1.3 Objetivos de la investigacion	20
1.3.1 Objetivos generales	20
1.3.2 Objetivos específicos	20
CAPITULO II	21
REVISIÓN DE LITERATURA.....	21
2.1 El internet de las cosas (iot)	21
2.1.1 Definición de iot por la itu-t.....	24
2.1.2 Aplicaciones.....	24
2.1.3 Arquitectura de iot.....	25

2.1.4	Capas de iot.....	28
2.1.5	Protocolos para iot	29
2.2	Red fog o computacion en la nube.....	32
2.2.1	Arquitecturas y servicios básicos	34
2.2.2	Componentes de la computación en la nube	34
2.2.3	Modelos y arquitecturas de la computación en la nube	35
2.2.4	Beneficios de la computación en la nube.....	37
2.2.5	Computación en la nube y el internet de las cosas	39
2.3	Red de sensores inalámbricos (wsn).....	41
2.3.1	Esquema de una red wsn	44
2.3.2	Nodos que forman una wsn.	45
2.3.3	Ventajas y beneficios de elegir una red inalámbrica	45
2.4	Hardware y sensores.....	46
2.4.1	Módulo wifi	47
2.4.2	Placa arduino nano	49
2.4.3	Sensores.....	51
2.4.4	Software y plataformas	56

2.5	Hipotesis.....	61
2.5.1	Hipótesis general	61
2.5.2	Hipótesis específicos	61
2.6	Antecedentes	21
CAPITULO III		62
MATERIALES Y MÉTODOS		62
3.1	Materiales.....	62
3.1.1	Hardware	62
3.1.2	Software.....	62
3.2	Método	63
3.2.1	Tipo de estudio	63
3.2.2	Técnicas de procesamiento y análisis.....	64
3.2.3	Instrumento de recolección de datos	77
CAPITULO IV.....		78
RESULTADOS Y DISCUSIÓN.....		78
4.1	Resultados del diseño de red fog con internet de las cosas.....	78
4.2	Resultados del diseño del prototipo para el monitoreo ambiental.	81

4.3	Resultados monitorización del lago titicaca.....	82
4.3.1	Resultados de muestreo en un nodo	83
4.3.2	Conexión del nodo al servidor.....	83
4.3.3	Envío de datos al servidor.....	84
4.3.4	Comparacion y validacion de reportes.....	86
	CONCLUSIONES	92
	RECOMENDACIONES	93
	REFERENCIAS.....	94
	ANEXOS	96

ÍNDICE DE FIGURAS

Figura N° 1: Modelo de arquitectura IoT de la ITU-t.....	26
Figura N° 2: Modelo de arquitectura IoT.	27
Figura N° 3: Capas y API de IoT.	28
Figura N° 4: Distribución de protocolos para IoT desde dispositivos a procesos de negocio.....	31
Figura N° 5 : Protocolos de Conectividad IoT.	32
Figura N° 6: Los dispositivos y servicios IoT parte de la Nube.	34
Figura N° 7 : Participantes de una WSN.	41
Figura N° 8: Esquema de una WSN.....	44
Figura N° 9: Figura de una Mota.	45
Figura N° 10 : Dispositivo wifi232.....	48
Figura N° 11 Vista frontal y posterior de la placa Arduino R3.	49
Figura N° 12 Sensor de temperatura y humedad DHT11	51
Figura N° 13: Sonda pH	53
Figura N° 14: Sensor de turbidez	55
Figura N° 15: Mensaje de inicio del software de Arduino.....	57
Figura N° 16 Entorno de desarrollo PhpMyadmin	60

Figura N° 17 Bahía del Lago Titicaca.....	64
Figura N° 18: Diagrama simulado de Red FOG en Cisco Packet Tracer	66
Figura N° 19: Calculo de enlace con Radio Mobile Nodo 1 y servidor EPIE	67
Figura N° 20: Calculo de enlace con Radio Mobile Nodo 1 y Nodo 2	68
Figura N° 21: Esquema de prototipo de WSN.....	70
Figura N° 22: Plataforma de creación del servidor.....	72
Figura N° 23: Portal web de IoT Titicaca.....	73
Figura N° 24: Portal de monitoreo.....	74
Figura N° 25: Historial de datos	75
Figura N° 26: Registro de sensores	76
Figura N° 27: Generación de Reporte.....	76
Figura N° 28 : Radioenlace del servidor hasta el nodo	79
Figura N° 29 : Diseño de la red en Cisco Packet Tracer.....	81
Figura N° 30: Prototipo de una Mota.....	82
Figura N° 31: Proceso de conexión mediante comandos AT	84
Figura N° 32: Proceso de inicio de conexión al servidor	84
Figura N° 33: captura de pantalla monitor serial Arduino.....	85
Figura N° 34: Diagrama de envío de Datos	85

Figura N° 35 : Graficas en el portal www.iottiticaca.com	86
Figura N° 36: Reporte grafico de pH en la bahía del Lago Titicaca	87
Figura N° 37: Reporte de IMARPE-SEDE PUNO	88
Figura N° 38: Captura de pantalla monitor serial valores pH	89
Figura N° 39: Reporte IMARPE-SEDE PUNO bahía de Puno	89
Figura N° 40: Medición de pH de las aguas del Lago Titicaca	90
Figura N° 41: Reporte de sensor de turbidez	91
Figura N° 42: Reporte de turbidez en diferentes puntos	91

ÍNDICE DE TABLAS

Tabla N° 1: Visión general de las plataformas Cloud IoT.....	40
Tabla N° 2: Características técnicas de la placa Arduino Uno.	50
Tabla N° 3: Especificaciones técnicas del sensor DHT11.....	52
Tabla N° 4: Características del sensor pH	54
Tabla N° 5: tabla de valores del sensor de pH.....	54
Tabla N° 6: Especificación del sensor de turbidez	56
Tabla N° 7 : Características de enlace nodo 1 y servidor	67
Tabla N° 8: Características de enlace nodo 1 y servidor	68
Tabla N° 9: Direcciones Ip y Mac del nodo	80
Tabla N° 10: Descripción del prototipo de una Mota	82

INDICE DE ANEXOS

Anexo A: Lista de Comandos AT	97
Anexo B: Vista de enlace nodo 1 y servidor.....	98
Anexo C: Configuracion de sensores en arduino	99
Anexo D: diseño de enlace en Radio Mobile	102
Anexo E: Motas en el Lago Titicaca (malecón)	103

ÍNDICE DE ACRÓNIMOS

6

6LowPAN
Redes con IP versión 6 de área personal de bajo poder y bajo consumo, 25

A

API
Application program interface, 6, 9, 23, 55

F

Fog
Niebla, 12, 13, 16, 17, 63, 91

G

GPIO
Son pines genericos, cuyo comportamiento, tanto de entrada como de salida., 10, 44

H

HTTP
Hypertext Transfer Protocol, 10, 20, 55, 99

I

IoT
Internet of Things, 3, 6, 8, 10, 11, 13, 18, 19, 20, 22, 23, 24, 25, 26, 27, 28, 29, 34, 35, 43, 53, 58, 59, 63, 64, 71, 77, 78, 80, 89, 91

M

M2M
Machine-to-Machine, 10, 24, 77

O

OTP
One-Time Programmable o Programable una vez, 49

W

WSN
Redes de Sensores Inalámbricos, 3, 4, 6, 11, 13, 36, 37, 40, 41, 42, 43, 60

X

XMPP
Extensible Messaging and Presence Protocol, 20, 56

RESUMEN

Esta investigación propone el diseño de una red FOG basado en internet de las cosas IoT, para monitorear la contaminación en la bahía del Lago Titicaca, monitoreo de las variables ambientales: temperatura, acidez y algunos parámetros de contaminación del Lago Titicaca, desde un abordaje enfocado al concepto de la “Internet de las Cosas”, Computación en la Nube, y Redes de Sensores Inalámbricos (WSN). La red consiste en unir de manera remota sensores en la bahía del lago mediante radio enlaces todas fijadas en el Lago con capacidad de registrar eventos de contaminación en las aguas del Lago Titicaca, haciendo el correcto uso de las Tecnologías de la información y la comunicación (TIC), la red permitirá reducir tiempo de análisis, optimizar costos, mejorar la calidad de los servicios de obtención de datos en tiempo real , disminuir riegos y ampliar la cobertura de monitoreo del Lago Titicaca para mitigar efectos de contaminación.

PALABRAS CLAVES: Contaminación, Internet de las cosas, Lago Titicaca, Red Fog.

ABSTRACT

This research proposes the design of an Internet-based FOG network of things IoT, to monitor pollution in the bay of Lake Titicaca, monitoring of environmental variables: temperature, acidity and some pollution parameters of Lake Titicaca, from a focused approach to the concept of the "Internet of Things", Cloud Computing, and Wireless Sensor Networks (WSN). The network consists of remotely linking sensors in the bay of the lake through radio links all set in the lake with the capacity to record pollution events in the waters of Lake Titicaca, making the correct use of information and communication technologies (TIC), the network will reduce analysis time, optimize costs, improve the quality of data collection services in real time, reduce risks and expand the monitoring coverage of Lake Titicaca to mitigate pollution effects.

Keywords: Pollution, Internet of things, Lake Titicaca, Red fog.

CAPITULO I

INTRODUCCIÓN

El Internet del Todo, Internet de las cosas es la evolución del internet que tiene la capacidad para reunir, analizar y distribuir datos que podemos convertir en información, conocimiento y en última instancia sabiduría; en este contexto IoT se vuelve inmensamente importante, especialmente en el contexto que se pretende utilizar en la presente investigación que nos ayuda a tener un entorno ambiental más sano del Lago Titicaca.

La presente tesis titulada “Diseño de una red FOG basado en internet de las cosas para monitorear la contaminación en la bahía del Lago Titicaca” el proyecto se desarrolla para brindar una alternativa fiable en cuanto a monitorización de datos en tiempo real.

El presente trabajo se desarrolla una red de sensores basado en internet de las cosas que realice la monitorización de turbidez, pH, temperatura del agua en la bahía de puno del Lago Titicaca, El desarrollo de la red de sensores comprende los subsistemas Middleware, Service Web y una interfaz gráfica de

usuario para monitorizar algunos parámetros de contaminación en la bahía de Lago Titicaca.

El procesamiento y disposición de los datos obtenidos permitirán que se conozcan cuantitativamente los niveles de concentración en la zona y su evolución; y que se disponga de información periódica en las zonas críticas de la bahía de Puno y su alcance comprende la integración a nivel técnico del sistema desarrollado basado en la arquitectura del Internet de las cosas.

1.1 PLANTEAMIENTO DE PROBLEMA

1.1.1 FORMULACION DEL PROBLEMA

La contaminación y el progresivo deterioro de las aguas de la bahía interior de Puno, constituye uno de los problemas más críticos que afecta el ecosistema del Lago Titicaca. En los últimos años se ha visto incrementada la contaminación como consecuencia del crecimiento poblacional de la ciudad de Puno, siendo que el 80 % de las aguas servidas recolectadas son tratadas en la laguna de estabilización Espinar y el restante 20 % es descargado en forma directa al Lago Titicaca sin ningún tratamiento. El sistema de tratamiento de aguas servidas presentan limitaciones en la remoción de Nitrógeno y Fosforo (N y P) debido a las características propias de dicho sistema, el mismo que está provocando la eutrofización de la Bahía Interior del Lago Titicaca. (PELT.)

Debido a que la bahía interior de Puno, se encuentra afectado por diversas actividades que realiza el hombre, el equilibrio del ecosistema presente en el Lago Titicaca se ve afectado, esta situación amerita y considera la ejecución de un Plan de Monitoreo de la contaminación y eutrofización en la Bahía Interior

de Puno a fin de disponer de información actualizada respecto a las condiciones ambientales de dichos ecosistemas que faciliten una acertada toma de decisiones posteriores, además ayuden a la gestión de descontaminación y recuperación de la bahía interior del Lago Titicaca en el mediano y largo plazo.

Esta problemática se origina en la falta de información permanente en el monitoreo del Lago Titicaca, la contaminación de la bahía del Lago Titicaca.

1.2 JUSTIFICACION DEL PROBLEMA

El proyecto de investigación “Diseño de una red Fog e internet de las cosas para monitorear la contaminación en la bahía del Lago Titicaca” será una red de transmisión de datos inalámbrico y posterior publicación o disponibilidad a través de computación en la nube o Red Fog para las instituciones responsables del monitoreo y mitigación ambiental el estudio se hará en los puntos críticos de contaminación en la bahía del Lago Titicaca.

Se justifica porque la contaminación y el progresivo deterioro de las aguas de la bahía interior de Puno, constituye uno de los problemas más críticos que afecta el ecosistema del Lago Titicaca. En los últimos años se ha visto incrementada la contaminación como consecuencia del crecimiento poblacional de la ciudad de Puno.

Actualmente la tecnología de Internet de las cosas permite interconectar cualquier objeto de nuestro medio, con esa premisa se plantea utilizar en el monitoreo de las aguas del Lago Titicaca haciendo una red de sensores, permitiendo compartir datos y brindar acceso de resultados muestreados a los usuarios.

Haciendo uso de red Fog internet de las cosas se pueden desarrollar infinidad de usos uno de ellos la monitorización de puntos sensibles de contaminación en la bahía del Lago Titicaca, el acceso a los datos de monitorización en tiempo real a través de la red Fog reducirá los costos de operación en tomar muestras como se hace convencionalmente.

Los beneficios que nos brindan este tipo de redes son muy interesantes puesto que podemos conectarnos a través de una plataforma virtual y ver los datos a través de diferentes dispositivos de forma rápida y eficiente sin hacer trámites burocráticos que se encarga de brindarnos ese tipo de información como el proyecto especial binacional Lago Titicaca PEBLT.

1.3 OBJETIVOS DE LA INVESTIGACION

1.3.1 OBJETIVOS GENERALES

Diseñar una red FOG con Internet de las cosas aplicada en el monitoreo de la contaminación ambiental en la bahía del Lago Titicaca.

1.3.2 OBJETIVOS ESPECÍFICOS

- a) Diseñar una red FOG con internet de las cosas.
- b) Implementar un prototipo en dos puntos críticos de la bahía del Lago Titicaca – Puno
- c) Monitorizar la contaminación del Lago Titicaca.

CAPITULO II

REVISIÓN DE LITERATURA

2.1 ANTECEDENTES

Eduardo Arias Monche en su investigación titulada “Demostración IoT-Cloud en tiempo real” en el año 2016, para optar el grado de master universitario de ingeniería de telecomunicación, en la Universidad abierta de Catalunya, UOC afirma que durante el desarrollo de este proyecto se ha hecho un análisis de situación actual de IoT con respecto a los servicios en la nube. Se ha mostrado su arquitectura, protocolos y algunos de los servicios más importantes en la actualidad.

Esta interacción con los servicios en la nube tiene un coste, de ahí que se haya utilizado como intermediario un sistema de computación en la niebla o FOG Computing, que interactúa entre los dispositivos y la ´ niebla, proporcionando la capacidad de decidir qué información se debe enviar a la nube de forma que se puedan reducir estos costes.

Joel Orlando Aragon Valladares, Diseño e implementación de una plataforma de gestión de una red de sensores aplicada a la monitorización de la calidad ambiental en la cuenca del río napo. Tesis para optar el grado de Ingeniero de Telecomunicaciones, de la PUCP en el año 2013, Los objetivos de desarrollar un sistema de información para las redes de sensores de calidad de agua y de aire y mostrar el comportamiento periódico de la data recolectada por los sensores almacenados en una base de datos a través de una interfaz Web.

En el artículo "Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN." De Alexander Borger 2010. Proporciona una descripción y las características generales de la red ubicua sensor (USN) y aplicaciones de USN y los servicios. También se analizan los requisitos de servicio de las aplicaciones y servicios de USN, y especifica los requisitos de capacidad NGN prolongados o nuevos basados en los requisitos de servicio.

En su trabajo de investigación de Francisco David Salgado Castillo y David Santiago Coello Moncayo, "Prototipo de monitoreo ambiental aplicando el Internet de las cosas con Arduino y Cloud Computing" para su graduación de ingeniero de Sistemas y Telecomunicaciones, Universidad de Azuay Cuenca Ecuador en el año 2012, su trabajo de titulación consiste en el desarrollo de un prototipo Electrónico para el monitoreo de las variables ambientales: temperatura, humedad, ruido y concentración de CO, desde un abordaje enfocado al concepto de la "Internet de las Cosas", Computación en la Nube, y Redes de Sensores Inalámbricos (WSN). El prototipo está realizado con la placa de hardware libre Arduino que se comunica por medio de Ethernet y WiFi con el internet a la plataforma ThingSpeak para el procesamiento de los datos en la

nube. Finalmente, se realiza la gestión de la información que permite mostrar 98 los gráficos de monitoreo correspondientes en un navegador web.

2.2 EL INTERNET DE LAS COSAS (IOT)

El Internet de las Cosas, IoT por sus siglas en inglés (“Internet of Things”), ha recibido varias denominaciones pero se puede describir como una red global de dispositivos inteligentes que pueden “sentir” e interactuar con su ambiente empleando el internet para su comunicación e interacción con usuarios y otros sistemas. (Doukas, 2012).

El uso de la palabra “Internet” en el término de IoT hace referencia a una metáfora: en la misma manera que las personas utilizan la web hoy en día, en un futuro las cosas también se comunicarán entre ellas, usarán servicios, proveerán datos y generarán valor agregado. La idea del Internet de las Cosas sugiere que, en lugar de tener un pequeño número de dispositivos de computación muy poderosos como laptops, teléfonos inteligentes, tabletas, etc., se podría tener un gran número de aparatos u objetos que sean “menos poderosos”, objetos como un par de zapatos, un espejo, un paraguas, un brazalete, una silla, etc., que monitoreen, registren y procesen datos relacionados a lo que sucede a su alrededor, su interacción con el medio ambiente, etc. Esta información puede provocar un mayor impacto en nuestras vidas que el que normalmente se obtendría con los dispositivos convencionales mencionados anteriormente. (McEwen, 2014)

2.2.1 DEFINICIÓN DE IOT POR LA ITU-T

La ITU-T define al internet de las cosas como una infraestructura de red global dinámica con capacidades auto configurables, basadas en protocolos de comunicación estándares e interoperables en donde las “cosas” físicas y virtuales:

- Tienen identidades, atributos físicos y personalidades virtuales,
- Utilizan interfaces inteligentes y
- Se integran a la perfección en la red de información. (ITU-T, 2017)

2.2.2 APLICACIONES

Las aplicaciones que se dan en los últimos años es variada e imprescindible para la sociedad actual, McEwen y Hakim mencionaban a ciertos “objetos encantados”, refiriéndose a cómo los dispositivos conectados, o cosas que, más allá de sus especificaciones funcionales, tienen capacidades de comunicación y procesamiento mucho mayores a la necesidad de una lámpara ordinaria, un paraguas o un espejo. Las aplicaciones de los componentes y dispositivos IoT pueden servir a satisfacer los requisitos planteados en situaciones como:

- Gestión de desperdicios
- Planificación urbana
- Sensores ambientales
- Domótica
- Herramientas digitales o Gadgets de interacción social
- Respuesta a emergencias
- Compras inteligentes

- Automatización en el hogar
- Muchas otras (Institute, Alexandra, 2011)

Como declara Charalampos Doukas en su libro, “La imaginación es el límite”. Y esa es otra razón por la que se considera que el Internet de las Cosas es algo realmente mágico.

2.2.3 ARQUITECTURA DE IOT

Se puede ver a la arquitectura de IoT de diferentes maneras, una de las cuales es comenzar con aplicaciones potenciales de ciudades inteligentes, transporte inteligente, edificios inteligentes, consumo de energía inteligente, etc. Esta arquitectura tiene diversos componentes. Las capas se pueden identificar como tecnologías específicas, y plantearemos opciones para identificar cada componente. También veremos capas que afectan tanto horizontal como verticalmente, como la gestión de acceso e identidad. (Beltrán, 2016)

Las capas son:

- Comunicaciones externas – Portal web, dashboard y APIs
- Procesamiento y analítica de datos
- Agregación / Bus – ESB y message bróker
- Comunicaciones – MQTT/ HTTP/ XMPP/ CoAP/ A Ytfgd MQP, etc.
- Dispositivos

Las capas transversales son las siguientes:

- Device Manager
- Control de acceso e identidades

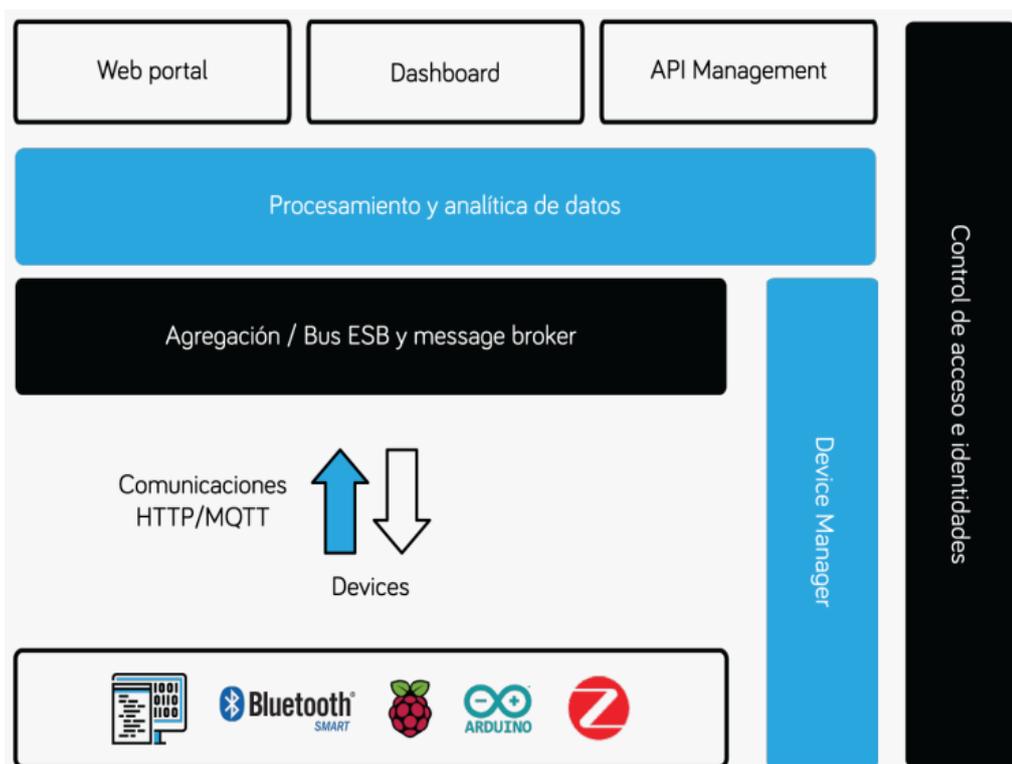


Figura N° 1: Modelo de arquitectura IoT de la ITU-t

Fuente: (Beltrán, 2016)

En cada bloque de aplicaciones potenciales se incluye la palabra “Smart” o “inteligente”. Este término hace referencia, en este caso, a la capacidad de conectarse a una red. Para soportar algunas o todas estas capacidades, se necesitan aplicaciones específicas para las “cosas” (capa de aplicación) las cuales se construyen sobre una variedad de diseños, incluyendo capacidades de gestión, capacidades de seguridad y muchos otros aspectos. Esto es solo un punto de vista en particular y existen muchos otros que están en trayectos paralelos, algunos coinciden y otros difieren en ciertos detalles. Las funciones que se encuentran debajo de la capa de aplicación, consisten en aquellas que se encargan de la práctica del problema, organización del servicio y de los datos. En contraste con aquellas que se encargan de las redes de comunicaciones, en

la base de la arquitectura, se asienta la capa de dispositivos, las “cosas” o dispositivos físicos.

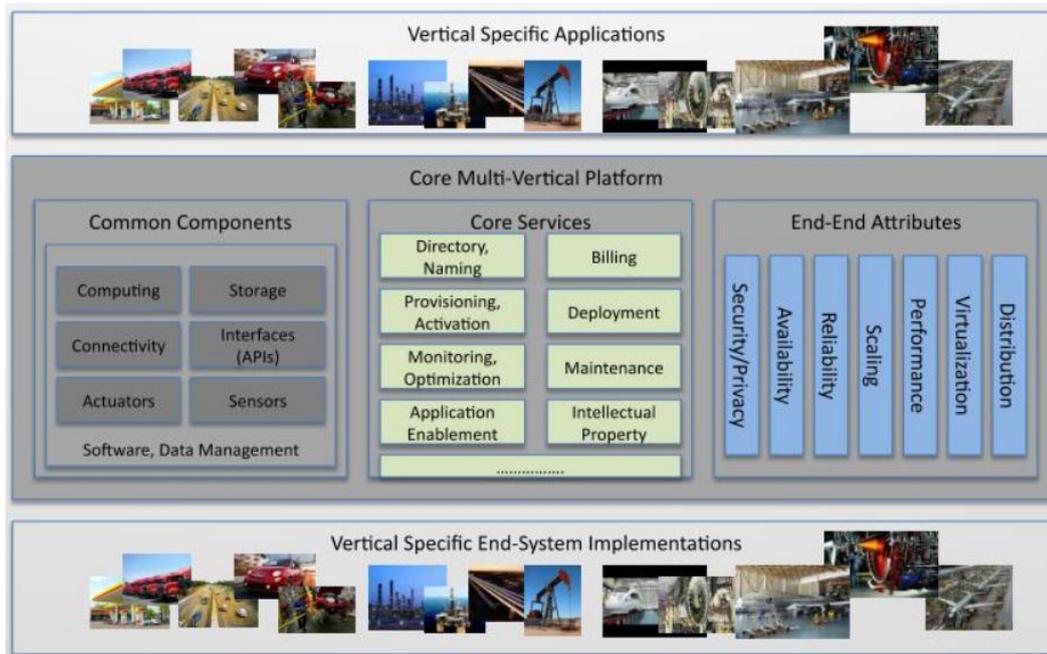


Figura N° 2: Modelo de arquitectura IoT.

Fuente: Chiang, 2015

En el otro modelo de arquitectura de la figura 2, uno más bien orientado a la parte comercial, también se tiene aplicaciones específicas verticales. Se puede observar una variedad de servicios que se ofrecen para soportar esas aplicaciones. En los componentes comunes se encuentran algunas de las características ya mencionadas, la habilidad de cómputo, almacenamiento, conectividad, etc. También se pueden observar todo tipo de servicios. Por ejemplo, la capacidad de nombrar a estas cosas, facturar y cobrar por los servicios, proveer diferentes servicios, monitorearlos y optimizarlos.

2.2.4 CAPAS DE IOT

En este caso, la manera en la que se definen las capas de trabajo es diferente a cómo se definen capas en la red comunicaciones en las que cada una sirve una función bien definida (capa física, de enlace, de red, de transporte, de aplicación). El término de capa se emplea en este aspecto para mostrar las separaciones de distinto modo de funcionamiento. Existe una capa de soluciones verticales que incluye ventas, de transporte, industrial, etc. Y debajo de ella, están algunos servicios creados por IoT, como por ejemplo DasS (Data as a service).

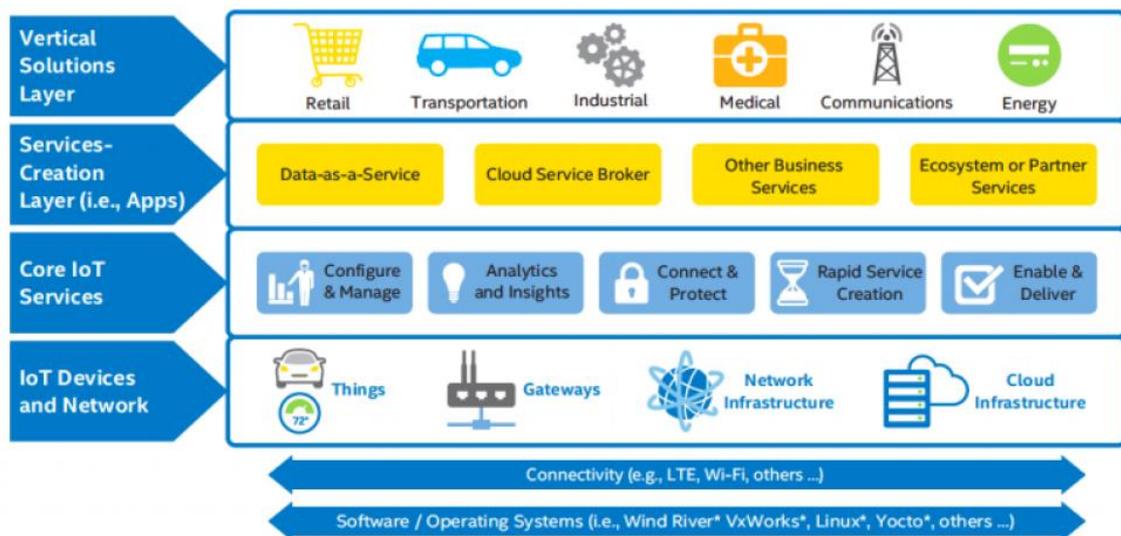


Figura N° 3: Capas y API de IoT.

Fuente: Chiang, 2015

Más abajo, se tienen los servicios núcleo de IoT como configuración y gestión, análisis de los datos, aprovisionamiento de servicios y fijación de su precio, etc. A ésta le sigue la capa subyacente de red y de dispositivos IoT que incluyen todas las “cosas”. (Chiang, 2015)

Cada capa vertical está limitada por otra, los servicios proveídos por el módulo se ven reflejados en la capa superior. Existe una separación de intereses, es decir, una división del trabajo, y cada capa tendrá una visibilidad y un control limitados sobre lo que debería pasar en otra capa.

2.1.5. Necesidades IoT

Para que las aplicaciones se corran como IoT, hay varios conjuntos de necesidades que brevemente se clasifican en:

- Creación de redes
- Dispositivos
- ✓ Hardware
- ✓ Diseño UI/UX (interfaz de usuario/experiencia de usuario)
- ✓ Prototipos: Raspberry Pi, Arduino, etc.
- Servicios
- ✓ Aplicaciones
- ✓ Economía y finanzas

2.2.5 PROTOCOLOS PARA IOT

Existe una gran cantidad de protocolos de comunicación que pueden aplicarse para el Internet de las Cosas. Algunos de los más empleados son los siguientes. (Electronics., 2014)

- **Ethernet:** una red de área local (LAN) cableada con alta velocidad.
- **MQTT** (Message Queue Telemetry Transport), un protocolo usado para la comunicación machine-to-machine (M2M) en el "Internet of Things". Este protocolo está orientado a la comunicación de sensores, debido a que

consume muy poco ancho de banda y puede ser utilizado en la mayoría de los dispositivos empotrados con pocos recursos (CPU, RAM.).

- **Bluetooth:** Velocidad mucho menor pero con una comunicación inalámbrica y de corto rango. Por ejemplo, el teléfono comunicándose con el automóvil.
- **802.11 WiFi:** Probablemente el protocolo de comunicación inalámbrica más común y fácil de encontrar en edificios, centros comerciales, universidades, etc.
- **Celular: 3G, 4G, LTE (Long-Term Evolution):** la red móvil o de celular, también inalámbrica pero con un rango de cobertura mucho mayor y con mejor soporte de movilidad. Para su despliegue debe pagarse por el uso del espectro radioeléctrico, a diferencia de WiFi que no tiene ningún costo por el uso de frecuencias. Algunos de los estándares de IoT dependen en gran medida de 4G y LTE y se han dado discusiones para hacerlos más livianos y permitan las comunicaciones entre las “cosas”.
- **802.15.4:** Que corresponde a las redes de área personal (PAN) como ZigBee de la ZigBee Alliance, MiWi de la empresa Microchip, etc.
- **6LowPAN:** Redes con IP versión 6 de área personal de bajo poder y bajo consumo. Para poder desplegar dispositivos que pueden ser usados en la vestimenta o incluso dentro del cuerpo humano.
- **PLC:** Comunicaciones por líneas de poder (Power Line Communication).
- **NFC:** Comunicaciones de campo cercano (Near Field Communication).
- **RFID:** Identificación por radiofrecuencia (Radio Frequency Identification)

Existen protocolos especializados para aplicaciones específicas.

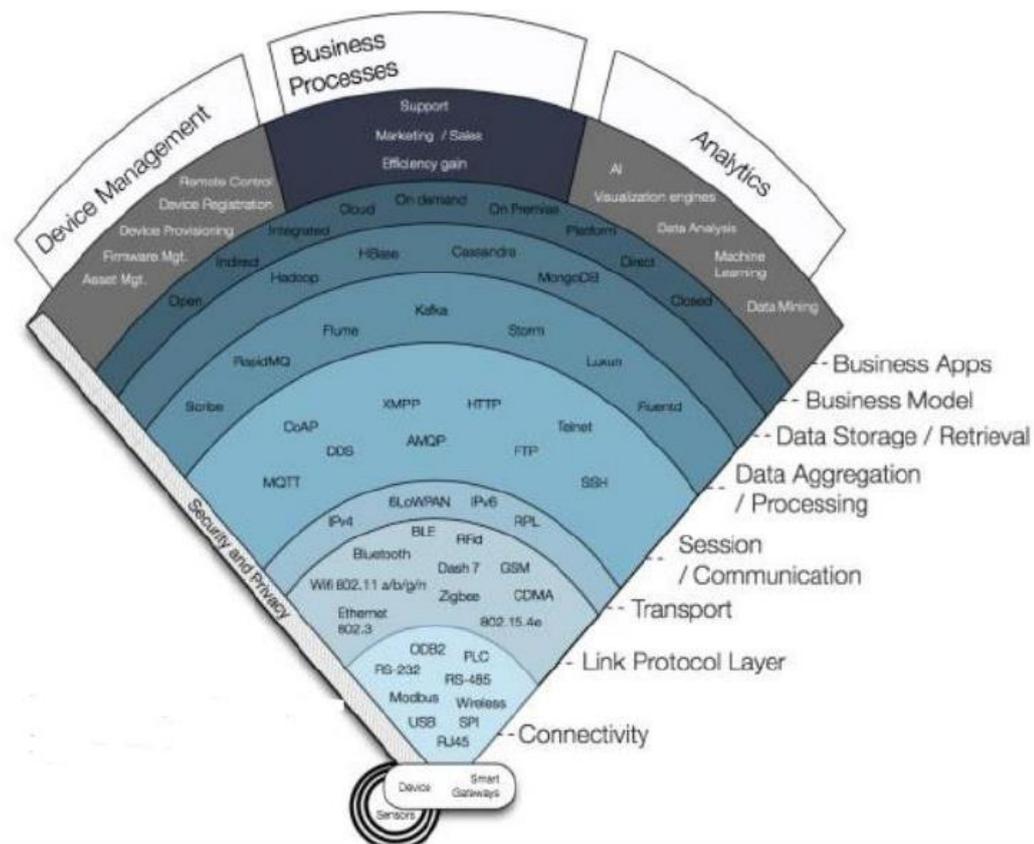


Figura N° 4: Distribución de protocolos para IoT desde dispositivos a procesos de negocio.

Fuente: Chiang, 2015

La figura 4, presenta el gran conjunto de protocolos comunes, algunos ya mencionados, para las correspondientes capas de enlace, física, transporte, red, sesión y aplicación. Sobre estas capas se trata más sobre datos (procesamiento, almacenamiento) y los protocolos que se utilizan para ello. Y por último los protocolos empleados para los servicios o para el modelo de negocios y sus aplicaciones (apps).

En la figura 6, muestra muchas de las variantes de protocolos juntas, pudiendo existir entre las “cosas” una comunicación WiFi y luego de pasar por la puerta de enlace o Gateway, se accede a una LAN por medio de Ethernet. De igual manera se puede tener un enlace móvil con 3G o 4G y otros protocolos correspondientes.

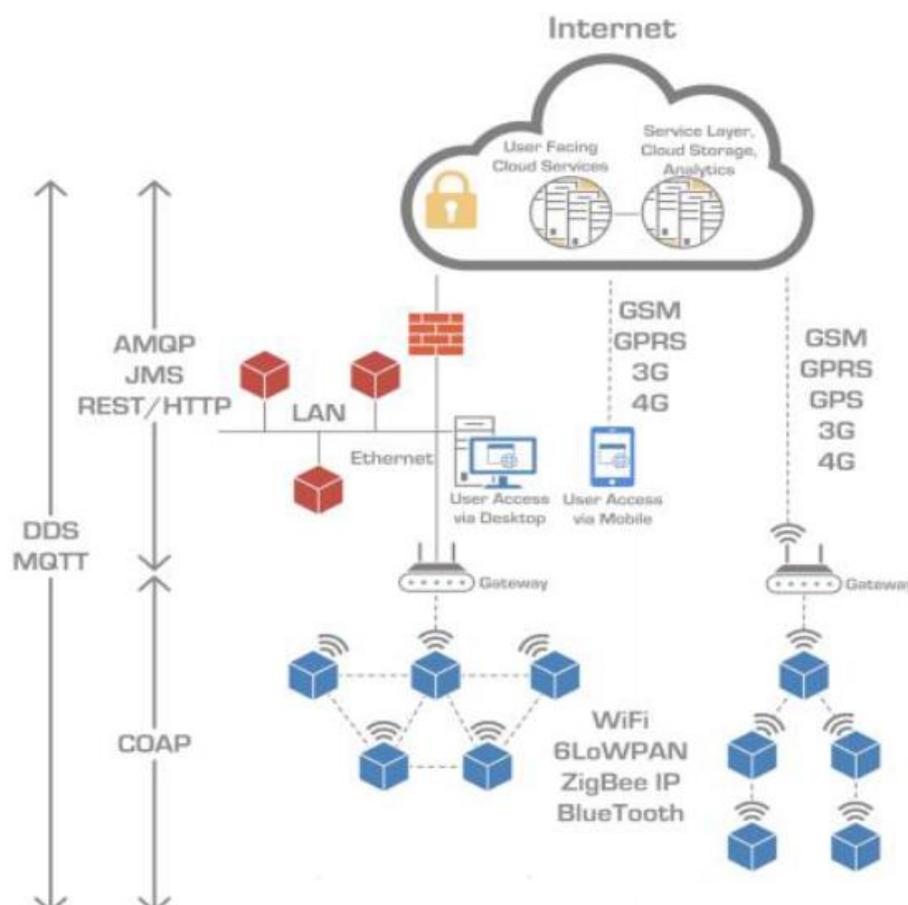


Figura N° 5 : Protocolos de Conectividad IoT.

Fuente: Chiang, 2015

2.3 RED FOG O COMPUTACION EN LA NUBE

Computación en la Nube o Cloud Computing es la entrega de información como un servicio en lugar de un producto, por lo cual se proveen los recursos, software e información compartidas a los computadores y a otros dispositivos como si fueran un servicio básico (como la red eléctrica) sobre una red (generalmente el internet). (Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., 2014)

En los últimos años se ha escrito mucho sobre el modelo de computación en la nube o Cloud Computing y dado que IoT suele requerir de dispositivos con

ciertas limitaciones, como el consumo de energía, hace que estos suelen tener como único propósito recoger información o realizar acciones, lo que los hace ideal para relegar el procesado, análisis y almacenamiento de la información a servicios disponibles en la nube (MONCHE, 2016)

La Computación en la Nube ofrece varios servicios, como los de computación, software, acceso a información y almacenamiento, los cuales no requieren de conocimiento del usuario sobre la ubicación física y la configuración del sistema que entrega los servicios. La figura 6 ilustra el concepto de Computación en la Nube y la comunicación entre los servicios (como aplicaciones, almacenamiento de información y recursos de procesamiento) y todo tipo de computación y dispositivos de comunicación.

La Computación de la Nube significa tener cada pieza de información que se necesita para cada aspecto de la vida diaria al alcance y lista para su uso. La información puede ser móvil, transferible y asequible al instante. La clave para tener un comportamiento portátil e interactivo, es la capacidad para sincronizar la información entre dispositivos, así como para acceder a información compartida. La información compartida es aquella a la que se accede en línea en cualquier lugar, tal como redes sociales, bancos, blogs, salas de prensa, comunidades pagadas, etc. (Hardy, 2016)

Una ilustración del concepto de la Computación en la Nube. Todo tipo de dispositivos de cómputo y comunicación pueden interactuar con la Nube y compartir los mismos recursos de información. Los dispositivos y servicios IoT son, de tal manera, una parte de la Nube. (Doukas, 2012)

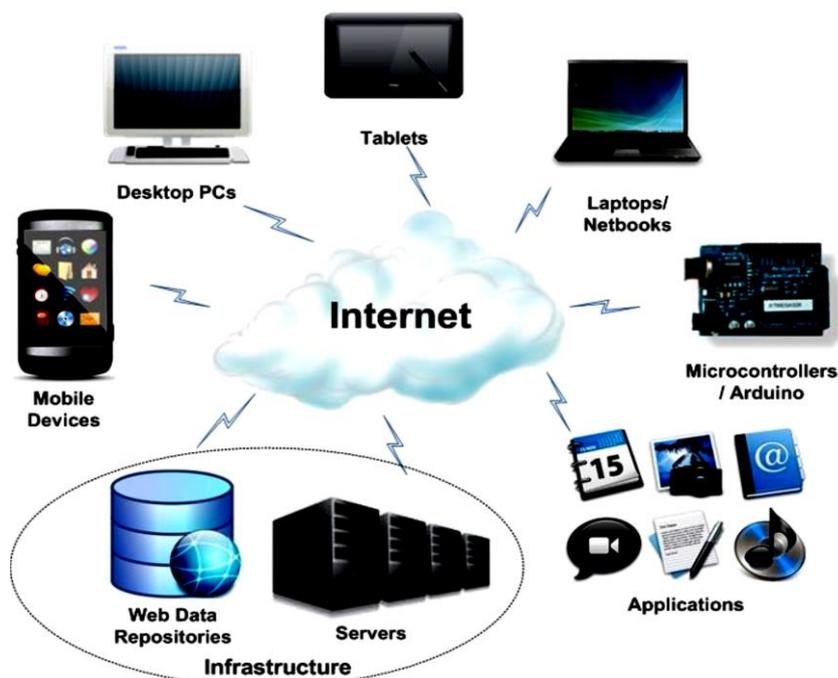


Figura N° 6: Los dispositivos y servicios IoT parte de la Nube.

Fuente: Doukas, 2012

2.3.1 ARQUITECTURAS Y SERVICIOS BÁSICOS

Un servicio Cloud o de la nube tiene tres distintas características que lo diferencian de las aplicaciones tradicionales y servicio de hosting. Se lo provee a demanda, es “elástico” (lo que significa que el usuario puede tener tanto o tan poco de un servicio como lo requiera en un momento dado) y finalmente, el servicio es gestionado totalmente por el proveedor. A continuación se describen los componentes principales y los modelos de la Computación en la Nube. (EDUCASE, 2017)

2.3.2 COMPONENTES DE LA COMPUTACIÓN EN LA NUBE

Un servicio de computación en la nube (como un servicio de almacenamiento en línea) consiste principalmente de tres componentes básicos: la aplicación, la plataforma y la infraestructura. (Doukas, 2012)

La Aplicación de la Nube es la principal interfaz con los usuarios. Esta puede ser una aplicación única (por ejemplo, un servicio de almacenamiento) o incluso un grupo de aplicaciones (como Google Docs) ejecutados e interactuados vía un navegador web, un escritorio local o un cliente remoto. Los usuarios no necesitan preocuparse de comprar e instalar ningún software (o incluso el sistema operativo). En su lugar, pagan el servicio en función de su uso

La Plataforma de la Computación en la Nube (en su mayoría los componentes apropiados de software que se utilizan) maneja cualquier tipo de operación de mantenimiento necesaria (incluyendo la activación e integración de nuevos recursos de hardware a la aplicación del usuario).

Esta puede considerarse como la capa intermediadora entre las aplicaciones alojadas en la nube y la infraestructura.

La Infraestructura de la Nube se refiere a la capa de hardware e incluye todos los componentes computacionales esenciales, de almacenamiento y de red que permiten la virtualización de los recursos y que permiten que funcione la plataforma de cómputo y provea los servicios correspondientes a los usuarios.

2.3.3 MODELOS Y ARQUITECTURAS DE LA COMPUTACIÓN EN LA NUBE

Hay varios tipos de nube disponibles, basados en la disponibilidad de sus recursos y en sus usos. Es decir, hay nubes públicas, comunitarias, híbridas y privadas. (EDUCASE, 2017)

- **Nube pública:** las nubes públicas describen a la computación de la nube en el sentido tradicional, donde los recursos son ofrecidos dinámicamente

al público por internet, vía aplicaciones web / servicios web, desde un proveedor que factura en base de servicios de computación.

- **Nube Comunitaria:** las nubes comunitarias comparten infraestructuras entre varias organizaciones de una comunidad específica con asuntos en común (seguridad, conformidad, jurisdicción, etc.), ya sea manejado internamente o por un tercero y alojado interna o externamente. Los costos se distribuyen entre menos usuarios que en una nube pública (pero más que en una nube privada), por lo que solo se consiguen algunos de los beneficios de la computación de la nube.
- **Nube híbrida:** la nube híbrida es una composición de dos o más nubes (privada, comunitaria o pública) que mantiene entidades únicas pero que están juntas, ofreciendo los beneficios de múltiples modelos de despliegue.
- **Nube privada:** opera su infraestructura solamente por una misma organización, ya sea manejada internamente o por un tercero y se aloja interna o externamente.

En cuanto a las arquitecturas y tipos de los servicios que ofrece un proveedor de la nube, principalmente hay tres tipos de servicios: (EDUCASE, 2017)

Infraestructura como un servicio: los proveedores de servicios se encargan de todo el costo del hardware esencial (servidores, equipamiento de red, almacenamiento y respaldos). Los usuarios solamente tienen que pagar para tener el servicio de computación. Y los usuarios construyen y manejan su propia aplicación de software. La infraestructura de Amazon EC2, es un buen ejemplo de este tipo de servicio.

Plataforma como un servicio-servicio: en este caso los proveedores, aparte del hardware, también proveen una plataforma específica o un conjunto de soluciones para el usuario (por ejemplo, un servidor de aplicaciones). Esto ayuda a los usuarios a ahorrar inversiones en hardware y software. El motor de aplicaciones de Google (Google APP Engine) y Jelastic provee este tipo de servicio. El gran beneficio de este servicio es que los usuarios necesitan enfocarse en el desarrollo y despliegue de su aplicación y no se preocupan de la configuración del sistema operativo y del servidor de la aplicación.

Software como un servicio: los proveedores de servicio usan su propio hardware e infraestructuras de plataforma para proveer a los usuarios aplicaciones de software específicas, como almacenamiento en línea, e-mail, aplicaciones de documentos, etc. Google Docs y Evernote son representantes de aplicaciones de software como servicio basadas en la nube.

2.3.4 BENEFICIOS DE LA COMPUTACIÓN EN LA NUBE

Entonces, ¿Por qué resulta necesaria la computación en la nube? y, ¿Por qué se busca almacenar datos de sensores en infraestructuras basadas en la nube? A continuación se describen algunas de las características y beneficios más importantes de la computación de la nube. (Doukas, 2012)

- **Eliminación o reducción de costos iniciales por el servicio de alojamiento web o hosting:** con las plataformas de la computación de la nube, no se necesita instalar uno mismo la infraestructura de hardware apropiada (que incluye servidores, Routers de red, firewalls, etc.) para alojar una aplicación en línea y almacenar datos de sensores.

- **Costos administrativos reducidos:** no se necesita gastar tiempo ni dinero en la administración (por ejemplo, configurar cosas, cuidar los respaldos, etc.) de los sistemas de software y hardware para la plataforma.
- **Mejor utilización de recursos:** en caso de que se requiera construir un servicio y proveerlo a otros usuarios; no es necesario dedicarse a construir sistemas (servidor web, servidor de base de datos) para cada servicio. Utilizando la tecnología de la nube, se puede reutilizar mucho del hardware existente para alojar varios servicios, incluso si es que éstos demandan diferentes recursos como sistemas operativos alternativos.
- **Escalabilidad demandada:** ¿cuánto almacenamiento o poder de procesamiento se necesita? ¿Se podría comenzar con un par de sensores para el monitoreo del hogar y luego invitar a otros usuarios a utilizar la misma plataforma y así expandirse a varias docenas de sensores? No es necesario extender el almacenamiento o los recursos de un servidor web propio; el proveedor del servicio de nube lo hace automáticamente.
- **Implementación rápida y fácil:** no se requiere conocer a profundidad sobre la configuración y despliegue de una aplicación basada en la web, instalar o configurar un servidor web, el sistema de la base de datos, hacer las conexiones apropiadas ni configurar varias instancias para cumplir ciertos requerimientos de escalabilidad. El usuario solamente se concentra en la aplicación e información que necesita para ser alojada en la nube.
- **Calidad de servicio y disponibilidad garantizada:** el proveedor de servicio de nube siempre garantiza la disponibilidad de hardware y

software. No hay que preocuparse por fallas de hardware, cortes de energía ni conexiones de red del lado del servidor.

- **Acceso en cualquier lugar:** lo más importante, los recursos de la nube (aplicaciones web, elementos de almacenamiento) pueden ser accedidos desde cualquier tipo de dispositivo computacional que tenga acceso al internet (incluyendo plataformas de microcontrolador, como Arduino).
- **Recuperación de desastre/respaldos:** no hay que preocuparse de respaldar información o crear mecanismos de recuperación. El proveedor se encarga de todo.

2.3.5 COMPUTACIÓN EN LA NUBE Y EL INTERNET DE LAS COSAS

Es claro que la Computación en la Nube provee grandes beneficios para aplicaciones alojadas en la web que también tienen requerimientos computacionales y de almacenamiento especiales. Además, ofrece maneras fáciles de acceder a ellas desde las redes y permite a los usuarios enfocarse en el desarrollo de aplicaciones. Por lo tanto, es bastante razonable que la mayoría de las plataformas IoT existentes dependan de infraestructuras en la nube. Éstas ofrecen la misma experiencia al usuario al estar usando/construyendo o desplegando aplicaciones en la nube. Los usuarios no tienen que preocuparse del mantenimiento de aplicaciones, recuperación, escalabilidad ni de asuntos de red. Ellos solamente necesitan configurar sus sensores para enviar información a través de plataformas IoT. (Doukas, 2012)

La siguiente sección presenta los servicios abiertos, basados en la nube, más difundidos para construir redes IoT y sus aplicaciones.

- **Lista de plataformas disponibles**

El número de plataformas disponibles que caracteriza el registro de datos en la nube para dispositivos de sensor está creciendo continuamente. Existe un gran número de ellas que la presentación de cada una podría representar un trabajo totalmente nuevo, pero que valdría la pena explorar. La siguiente lista mostrada en la tabla N° 1 hace un resumen de algunas de plataformas IoT.

Tabla N° 1: Visión general de las plataformas Cloud IoT.

Nombre	Características	Free Open Source	URL
THINGSPEAK	Visualiza y almacena datos de sensores en línea	Sí Sí	http://www.thingspeak.com
NIMBITS	Registro de datos en la nube	Sí Sí	http://www.nimbits.com
IDIGI	Plataforma en la nube para dispositivos	Sí No	http://www.idigi.com
SENSORCLOUD	Visualiza y almacena datos de Sensores en línea.	Sí No	http://www.sensorcloud.com
OPEN.SEN.SE	Plataforma para el Internet del Todo	Sí No	http://open.sen.se
EXOSITE	Plataforma y Portales de datos basados en la nube y gestión de dispositivos	Sí No	http://www.exosite.com
EVRYTHNG	Motor de Software y Plataforma de la Nube	Sí No	http://www.evrythng.net
PARAIMPU	Herramientas sociales para las cosas	Sí No	http://paraimpu.crs4.it
MANYBOTS	Recolecta y gestiona información desde varios dispositivos	Sí No	http://www.manybots.com
LELYLAN	Orientado en automatización del hogar y monitoreo	Sí No	http://www.lelylan.com

FUENTE: Doukas, 2012

2.4 RED DE SENSORES INALÁMBRICOS (WSN)

Una WSN está conformada de nodos, Gateway y software, los nodo interactúan con los sensores y mediante un módulo instalado en ella, es posible la configuración del nodo para manejar su comportamiento y monitoreo; lo cual es posible mediante los nodos programables. Dicho esto, se puede determinar que una WSN está constituida por sistemas autónomos desplegados de forma densa y aleatoria para recolectar información sobre un fenómeno en particular como detección de eventos como incendios, lluvias, inundaciones, etc. Además de mediciones periódicas como agricultura de precisión, aproximación de una función como temperatura y tiempo. (National Instruments, 2017).

Una red WSN se diseña con el propósito de monitorear un lugar o equipo en específico, mediante una infraestructura de nodos, con la finalidad de control o análisis local de cada una de estas redes. Se puede armar con una infinidad de nodos para su uso dependiendo del tipo de trabajo que se le quiera dar. Los nodos se pueden comunicar entre sí, por cableado o por conexión inalámbrica. En la Figura 7, Se puede observar cuáles son los participantes de una Red WSN.

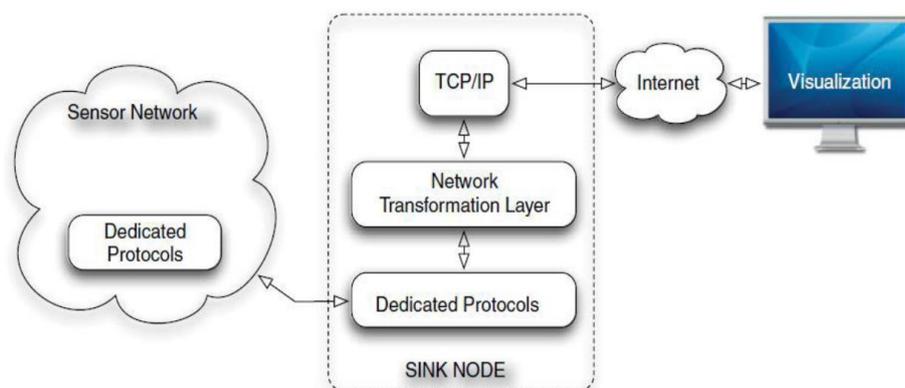


Figura N° 7 : Participantes de una WSN.

Fuente: Samaniego Armijos

A una red WSN se la puede desplegar de diferentes formas, la primera es por modo aleatorio donde se puede esparcir los nodos por medio aéreo y así crear la red; el despliegue regular por ubicación planificada y nodos móviles donde se puede complementar el despliegue deficiente o con movimiento pasivo. (National Instruments, 2015).

En la implementación de la red se puede obtener los siguientes servicios:

Medio Ambiente:

- Detección de fuego en bosques.
- Mapa de bio complejidad del ambiente (monitoreo de árboles).
- Detección de fluidos (lluvias, nivel de aguas en ríos y sensores de clima).
- Agricultura de precisión (monitorear el nivel de pesticidas, nivel de erosión del suelo y contaminación ambiental).

Salud:

- Tele-monitoreo de datos psicológicos humanos.
- Monitorear el comportamiento de personas adultas mayores, en una caída.
- Determinar el comportamiento del corazón, presión sanguínea.
- Localizar y monitorear pacientes y doctores dentro de un hospital.
- Administración de fármacos en un hospital.
- Identificación de alergias mediante sensores, determinando la medicación requerida de los pacientes.

Residencial y Comercial:

- Automatización de residencias y ambientes inteligentes.

- Colocación de sensores en microondas, cocinas, refrigeradoras, para interactuar con las redes externas.
- Museos interactivos
- Control de inventario
- Detección y localización de vehículos
- Detectar y monitorear vehículos robados

El Gateway recibe los datos obtenidos de manera inalámbrica para operar independientemente, conectándose al sistema donde el usuario puede obtener la información deseada, procesar y analizar los datos medidos usando software.

Factores a considerar en el Diseño:

- Tolerancia a fallos
- Escalabilidad
- Limitaciones del hardware
- Topología de red de sensores
- Entorno de operación
- Medio de transición
- Consumo de energía

Arquitectura De Protocolos

- Combina energía y enrutamiento.
- Integra protocolos de networking o de creación de redes.
- Comunica eficientemente por medios inalámbricos.
- Promueve el trabajo colaborativo entre nodos.

Para poder comprender un poco más sobre el sistema *Gateway* y su funcionamiento dentro de una red WSN, el *Gateway* es el encargado de recibir

los datos medidos desde los nodos distribuidos y desde los puentes a la red empresarial. En otras palabras, es el coordinador de red encargado de la autenticidad del nodo y almacenamiento de mensajes, aquí se reúne, analiza y presenta los datos de medida. En una red WSN pueden existir múltiples *gateways* donde cada uno se conecta y se comunica en un canal inalámbrico independiente y con la respectiva configuración de software, los cuales son monitoreados para obtener los resultados solicitados.

2.4.1 ESQUEMA DE UNA RED WSN

La figura 8, muestra un esquema sencillo de una WSN, la cual está conformada por procesadores o placas con comunicación por radio, y que en este contexto se denominan motes o motas (M). Además de sensores y hardware para adquisición de datos (S), y la interfaz de red o puerta de enlace, Gateway. El conjunto de la mota con los sensores forma un nodo de la WSN.

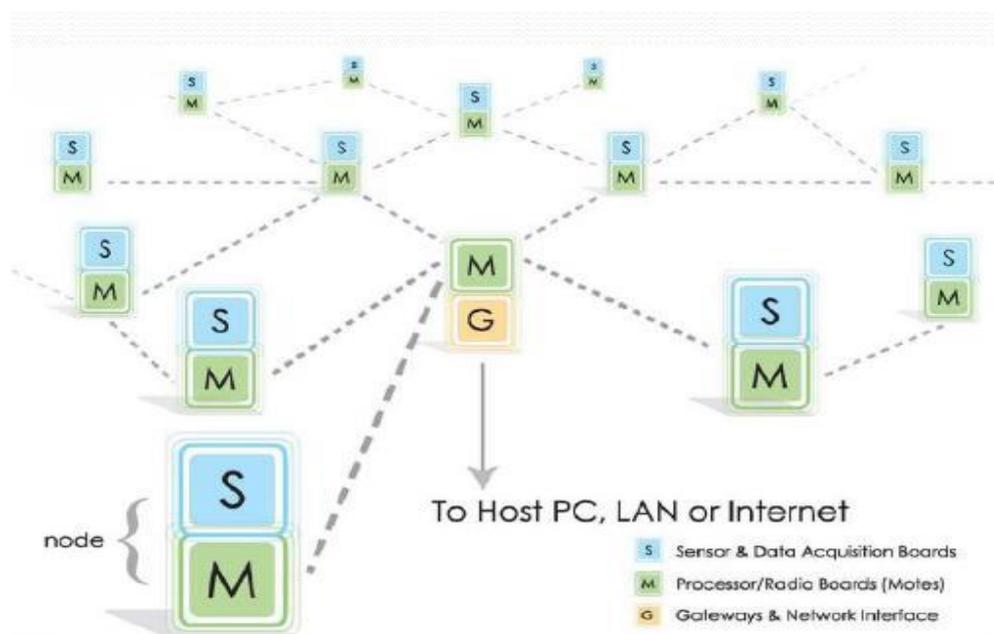


Figura N° 8: Esquema de una WSN.

Fuente: Samaniego Armijos, 2015

2.4.2 NODOS QUE FORMAN UNA WSN.

Una mota de un nodo dentro de una WSN, no es más que el hardware que realiza el tratamiento de los datos que son adquiridos por los sensores, los almacena si es necesario, y transmite esa información hacia la mota que posea la puerta de enlace hacia la red o al internet, para que dichos datos sean analizados y tratados. La figura 9 enseña la arquitectura básica de una mota.

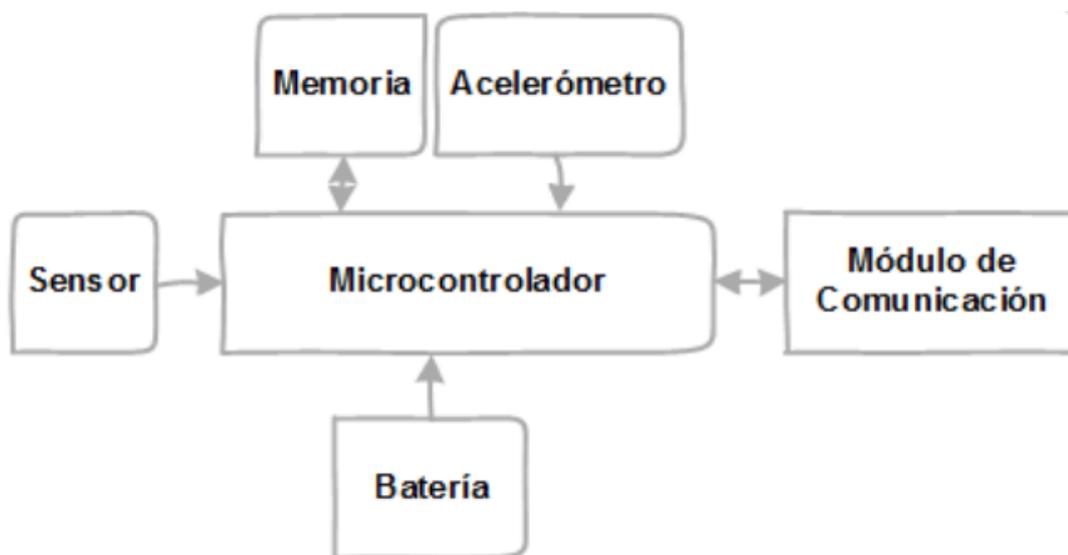


Figura N° 9: Figura de una Mota.

Fuente: (Armijos, 2015)

2.4.3 VENTAJAS Y BENEFICIOS DE ELEGIR UNA RED INALÁMBRICA

Una Red WSN tiene una estabilidad y fiabilidad de igual calidad que un sistema de red de medida normal a base de cableado, con la diferencia que una red WSN es más flexible, tiene costos más bajos, con la opción incluso para crear la red.

Estos sistemas están diseñados para soportar áreas de difícil acceso y lugares con temperaturas irregulares. Sin embargo, son capaces de trabajar

normalmente ante las dificultades ambientales que se presenten. Son sistemas intuitivos donde no se necesita conocimiento de programación embebida, es decir, son programables en su totalidad y están listos para trabajar en cualquier circunstancia; fácilmente se puede programar el nodo con aplicaciones embebidas y entornos gráficos. Se puede obtener la información de una manera sencilla mediante visualización y análisis en la web, accediendo a los datos desde cualquier lugar, en cualquier plataforma. Habilitada en tiempo real, móvil o laptop.

La facilidad de no tener limitado el sistema de red de manera inalámbrica, permite tener la mayor flexibilidad, monitorear lugares de difícil alcance, incluso de ser necesario, reubicar los nodos o también añadir nodos a la red conforme la aplicación vaya creciendo.

En consecuencia, al evitarse utilizar cableado de cobre y Ethernet, se obtiene una mejor economía en el proyecto al ahorrar gran cantidad de dinero, ganando tiempo de trabajo, obteniendo condiciones ambientales y condiciones de los equipos, al trabajar con los sistemas inalámbricos al evitar la producción de tiempos muertos y reducir los costos de mantenimiento (National Instruments, 2015)

2.5 HARDWARE Y SENSORES

En este capítulo, se presenta el hardware necesario para realizar el prototipo propuesto de Monitoreo Ambiental. Este es el hardware que formaría parte de una red WSN y podría emplearse en distintas aplicaciones IoT.

Se describe la mota o plataforma de microcontrolador a utilizarse, Arduino, con sus shields o módulos Ethernet y WiFi. Finalmente, se indican las características

técnicas de los sensores que medirán las variables ambientales de temperatura, humedad, PH, concentración de sólidos (turbidez).

2.5.1 MÓDULO WIFI

Este módulo incluye todo lo necesario para conectarse a un punto de acceso WIFI mediante comandos de texto AT, vía una puerta serie, que puede ser configurada a diferentes velocidades. Una vez que lo programamos para conectarse a nuestra red WIFI, el módulo es capaz de enviar información que le remitimos vía la puerta serie a una dirección IP y puerto que deseemos. Cuando se trata de recibir, limpia todo el empaquetado TCP/IP y nos reenvía por la puerta serie la información de datos limpia de polvo y paja, con lo que tiene la enorme virtud de permitirnos olvidarnos de la gestión del TCP/IP y de las demandas de procesador y memoria que suponen. A cambio no es exactamente una conexión WIFI, porque no tenemos acceso al stack o al socket IP pero para el Arduino esto es casi una ventaja.

Con una experiencia bastante más avanzada, más que nada en programación se puede hacer uso del microcontrolador que tiene integrado este poderoso módulo, el cual posee 16 GPIO (General Purpose Input Output), los cuales se pueden configurar como entradas o salidas, al igual que con Arduino, como interrupciones para sacar al módulo de modo sleep, y hasta posee un ADC. Los 16 pines, son en total los que posee este microcontrolador, por lo tanto, por más que podemos darles las funciones antes mencionadas, en algunos de ellos también se encuentran la comunicación SPI, UART, entre otras funciones.



Figura N° 10 : Dispositivo wifi232

Fuente: www.waveshare.com

Características Técnicas

- Protocolos soportados: 802.11 b/g/n
- Wi-Fi Direct (P2p), Soft Access Point
- Stack TCP/IP integrado
- PLL, reguladores y unidades de manejo de energía integrados
- Potencia de salida: +19.5dBm en modo 802.11b
- Sensor de temperatura integrado
- Consumo en modo de baja energía: <10 uA
- Wi-Fi 2.4 GHz, soporta WPA/WPA2
- Tamaño ultra reducido (11.5mm x 11.5mm)

- Conversor analógico a digital de 10-bit
- Soporta variedad de antenas
- Integrated low power 32-bit MCU
- SDIO 2.0, SPI, UART, I2C
- Rango de operación $-40^{\circ} \sim 125^{\circ}$

2.5.2 PLACA ARDUINO NANO

El Arduino Nano es un tablero pequeño, completo y compatible con el tablero basado en el ATmega328 (Arduino Nano 3.x). Tiene más o menos la misma funcionalidad del Arduino Duemilanove, pero en un paquete diferente. Le falta solo un conector de alimentación de CC, y funciona con un cable USB Mini-B en lugar de uno estándar

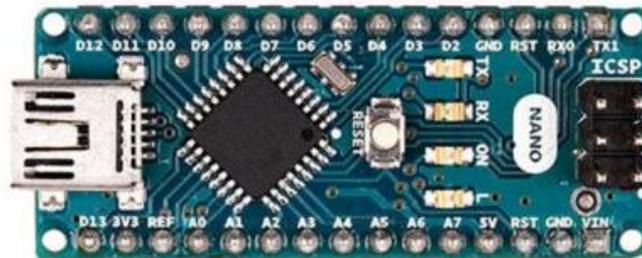


Figura N° 11 Vista frontal y posterior de la placa Arduino R3.

Fuente: www.arduino.com

Características Técnicas

El Arduino nano es una placa electrónica basada en el ATmega328. Cuenta con 14 pines digitales de entrada / salida (de los cuales 6 se pueden utilizar como salidas PWM), 6 entradas analógicas, un resonador cerámico de 16 MHz, una conexión USB, un conector de alimentación, un header ICSP, y un botón de

reset. Contiene todo lo necesario para apoyar el Microcontrolador; simplemente hay que conectarlo a un computador con un cable USB o alimentarlo con un adaptador o una batería. La tabla muestra el resumen de características técnicas.

Tabla N° 2: Características técnicas de la placa Arduino Uno.

Características del nano Arduino.	
Microcontrolador	ATmega328
Voltaje de operación	5V
Voltaje de entrada (límites)	6-20V
Pines de E/S Digitales	14-6 proveen salidas PWM
Corriente DC por Pin de E/S	40 mA
Corriente DC por Pin	3.3V 50 mA
Memoria Flash	32 KB
SRAM	2KB
EEPROM	1KB
Velocidad de reloj	16 MHz

Fuente: Portal Arduino.com

Comunicación

El Arduino Nano tiene varias facilidades para comunicarse con una computadora, otro Arduino u otros microcontroladores. El ATmega328 proporciona comunicación serial UART TTL (5V), que está disponible en los pines digitales 0 (RX) y 1 (TX). Un FTDI FT232RL en la placa canaliza esta comunicación en serie a través de USB y los controladores FTDI (incluidos con el software Arduino) proporcionan un puerto virtual para el software en la computadora. El software Arduino incluye un monitor serie que permite el envío

de datos textuales simples hacia y desde la placa Arduino. Los LED RX y TX de la placa parpadearán cuando los datos se transmitan a través del chip FTDI y la conexión USB a la computadora (pero no para la comunicación serial en los pines 0 y 1). Una biblioteca de Software Serial permite la comunicación serial en cualquiera de los pines digitales de Nano. El ATmega328 también admite la comunicación I2C (TWI) y SPI. El software Arduino incluye una biblioteca Wire para simplificar el uso del bus I2C. Para usar la comunicación SPI, consulte la hoja de datos de ATmega328.

2.5.3 SENSORES

Para medir las variables ambientales propuestas; ruido, concentración de monóxido de carbono, temperatura y humedad; se requieren de sensores o transductores correspondientes. A continuación, se indicarán los sensores empleados en este prototipo de monitoreo ambiental.

Temperatura y Humedad

Se puede medir ambos de estos parámetros con un solo sensor, el sensor DHT11 mostrado en la figura 12

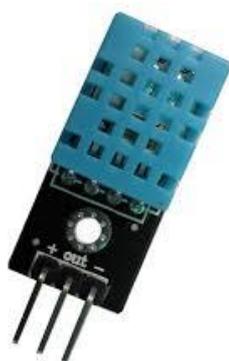


Figura N° 12 Sensor de temperatura y humedad DHT11

Fuente: www.waveshare.com

De acuerdo a su hoja de datos, este sensor de temperatura y humedad DHT11, dispone a la salida una señal digital calibrada. Su tecnología garantiza la confiabilidad y estabilidad a largo plazo. Posee un microcontrolador de 8-bit. Este sensor incluye un elemento de resistencia y detección de humedad de los dispositivos de medición de temperatura, NTC (Negative Temperature Coefficient o coeficiente de temperatura negativo).

Su descripción indica que cada sensor DHT11 presenta características de calibración precisas para la cámara de calibración de humedad. Los coeficientes de calibración están almacenados en la memoria de programa OTP (One-Time Programmable o Programable una vez), los cuales son utilizados por el proceso interno de detección de señales. El producto viene en un paquete de 4 pines de una sola fila. (D-Robotics, 2010)

Tabla N° 3: Especificaciones técnicas del sensor DHT11.

DESCRIPCION	CARACTERISTICAS
Voltaje de alimentación:	3 – 5.5 V CD
Rango de medición Humedad:	20 – 90 %HR
Temperatura	0 – 50 Celsius
Precisión Humedad	+ - 5 %HR
Resolución o sensibilidad Humedad	1 %HR;
Repetibilidad Humedad	+ -1 %HR;
Histéresis de humedad	+ - 1 % HR
Estabilidad a largo plazo	+ - 0.5 % HR/año
Período de detección Promedio:	2s
Dimensiones Tamaño	12*14.5*5.5mm

FUENTE: www.waveshare.com

SENSOR DE PH

Medidor de pH analógico, especialmente diseñado para controladores Arduino y tiene una conexión y características incorporadas, simples, prácticas y prácticas. Tiene un LED que funciona como el indicador de alimentación, un conector BNC y la interfaz del sensor pH 2.0. Para usarlo, simplemente conecte el sensor de pH con el conector BNC, y conecte la interfaz pH 2.0 en el puerto de entrada analógica de cualquier controlador Arduino. Si está pre programado, obtendrá el valor de pH fácilmente. Viene en una caja de plástico compacta con espuma para un mejor almacenamiento móvil.



Figura N° 13: Sonda pH

Fuente: www.waveshare.com (2017)

Para garantizar la precisión de la sonda de pH, debe usar la solución estándar para calibrarla regularmente. En general, el período es de aproximadamente medio año. Si mide la solución acuosa sucia, necesita aumentar la frecuencia de calibración.

Tabla N° 4: Características del sensor pH

DESCRIPCION	CARACTERISTICA
Potencia del módulo	5.00V
Tamaño del módulo	43 mm x 32 mm
Rango de medición	0-14PH
Medición de temperatura	0-60 °C
Precisión	0.1pH (25 °C)
Tiempo de respuesta	≤ 1min
Sensor de pH	conector BNC

Fuente: www.waveshare.com (2017)

La salida del electrodo de pH es Milivolts, y el valor de pH de la relación se muestra de la siguiente manera a una temperatura de 25 C° estas características son esenciales al momento de programar para dar valores exactos.

Tabla N° 5: tabla de valores del sensor de pH

Voltaje (mV)	pH valor	VOLTAGE (mV)	pH valor
414.12	0.00	- 414.12	14.00
354.96	1.00	-354.96	13.00
295.80	2.00	-295.80	12.00
236.64	3.00	-236.64	11.00
177.48	4.00	-177.48	10.00
118.32	5.00	-118.32	9.00
59.16	6.00	-59.16	8.00
0.00	7.00	0.00	7.00

Fuente: www.waveshare.com (2017)

Sensor de turbidez SKU: SEN0189

El sensor de turbidez Arduino de gravedad detecta la calidad del agua al medir los niveles de turbidez. Utiliza la luz para detectar partículas suspendidas en el agua al medir la transmitancia de la luz y la tasa de dispersión, que cambia con la cantidad de sólidos suspendidos totales (TSS) en el agua. A medida que aumenta el TTS, aumenta el nivel de turbidez del líquido.



Figura N° 14: Sensor de turbidez

Fuente: www.waveshare.com (2017)

Los sensores de turbidez se utilizan para medir la calidad del agua en ríos y arroyos, las mediciones de aguas residuales y efluentes, la instrumentación de control para el asentamiento de estanques, la investigación del transporte de sedimentos y las mediciones de laboratorio.

Este sensor de líquido proporciona modos de salida de señal analógica y digital. El umbral es ajustable cuando está en modo de señal digital. Puede seleccionar el modo de acuerdo con su MCU.

Tabla N° 6: Especificación del sensor de turbidez

DESCRIPCION	CARACTERISTICA
Voltaje de funcionamiento:	5V DC
Corriente de funcionamiento:	40mA (MAX)
Tiempo de respuesta:	<500ms
Resistencia de aislamiento:	100M (Min)
Salida analógica:	0-4.5V
Salida digital:	Señal de nivel alto / bajo (puede ajustar el valor umbral ajustando el potenciómetro)
Temperatura de funcionamiento:	5 °C ~ 90 °C
Temperatura de almacenamiento:	-10 °C ~ 90 °C
Peso:	30g
Dimensiones del adaptador:	38 mm * 28 mm * 10 mm / 1.5 pulgadas * 1.1 pulgadas * 0.4 pulgadas

Fuente: www.waveshare.com (2017)

2.5.4 SOFTWARE Y PLATAFORMAS

En esta investigación se expone sobre el software de programación de Arduino para el prototipo de monitoreo ambiental y de las plataformas de nube IoT seleccionadas, ThingSpeak y Nimbits, por ser las únicas Open Source y Free Source, también se puede elaborar un servidor propio basado en IoT a través de XAMPP que es un entorno libre y gratuito para este fin que a continuación se detallan.

- **Arduino IDE**

El entorno de desarrollo integrado de Arduino - o Arduino Software (IDE) - contiene un editor de texto para escribir código, un área de mensajes, una

consola de texto, una barra de herramientas con botones para funciones comunes y una serie de menús. Se conecta al hardware Arduino para cargar programas y comunicarse con ellos.



Figura N° 15: Mensaje de inicio del software de Arduino.

Fuente: Arduino Ide V.1.7.10

Los programas escritos utilizando el Software de Arduino (IDE) se llaman sketches o bocetos. Estos sketches se escriben en el editor de texto y se guardan con la extensión de archivo (.ino). El editor tiene funciones para cortar/pegar y para buscar/reemplazar texto. El área de mensajes proporciona retroalimentación mientras se guarda y exporta el programa, también muestra los errores que pudieron darse en estos procesos. La consola muestra la salida de texto por el software de Arduino (IDE), incluidos mensajes de error completos y otra información. La esquina derecha inferior de la ventana muestra el modelo de placa Arduino configurado y puerto serie al cual está conectado. Los botones

de la barra de herramientas permiten verificar y cargar programas, crear, abrir y guardar sketches, y abrir el monitor serial. (Arduino, s.f.)

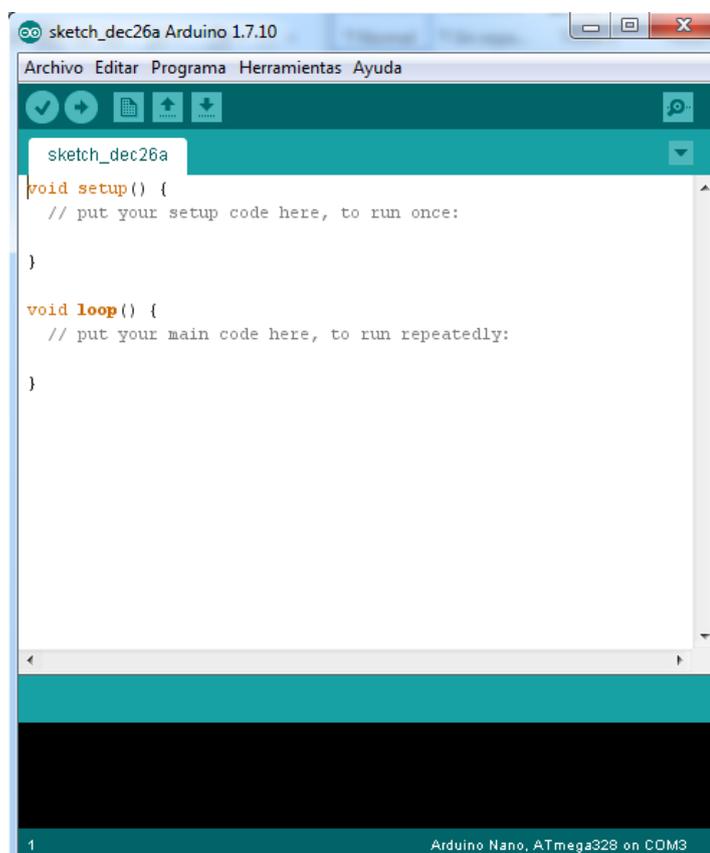


Figura 6.2. Interfaz del IDE, un sketch básico.

Fuente: Arduino Ide V1.7.10

- **ThingSpeak**

ThingSpeak es una plataforma para el "Internet de las cosas" la cual cuenta con una interfaz de programación de aplicaciones o API (application program interface) de código abierto para almacenar y recuperar datos de las cosas usando el protocolo de transferencia de hipertexto HTTP (Hypertext Transfer Protocol) a través de internet o a través de una red de área local. Con ThingSpeak, se pueden crear aplicaciones de registro de lecturas de sensores, aplicaciones para seguimiento y localización, y una red social de las cosas con actualizaciones de estado.

Además de almacenar y recuperar datos numéricos y alfanuméricos, la API de ThingSpeak permite el procesamiento de datos numéricos como escala de tiempo, promedios, mediana, sumatorias y redondeo. Cada canal ThingSpeak soporta entradas de datos de hasta 8 campos, además de la información geográfica del dispositivo de transmisión (la mota) como latitud, longitud y elevación. Los canales soportan los formatos JSON, XML y CSV para su integración en aplicaciones.

La aplicación ThingSpeak también cuenta con control de zonas horarias, gestión de la lectura / de claves y gráficas basadas en JavaScript. El Soporte para ThingSpeak está disponible en la Comunidad del sitio web que cuenta con un blog, foro, documentación y tutoriales. (Iobridge, ThingSpeak) además cuenta con una aplicación móvil para teléfonos Android para visualizar las gráficas de los datos que recibe en el canal.

Para empezar a utilizar la plataforma ThingSpeak, se sigue la siguiente guía de inicio rápido (ThingSpeak, s.f.):

- **NIMBITS**

Nimbits es un servicio de registro de datos y Plataforma para conectar sensores y software a la nube. Nimbits aporta para el Internet de las Cosas, proporcionando una plataforma que se puede construir localmente en sistemas embebidos, filtrar ruido, ejecutar reglas y enviar datos importantes hasta la nube.

El servidor de Nimbits registra y procesa datos y ejecuta reglas que se definen en base a la información recibida. Las reglas pueden ser cálculos, estadísticas, alertas de correo electrónico, mensajes XMPP (Extensible Messaging and Presence Protocol), notificaciones push (en dispositivos móviles) y más.

Es posible descargar el servidor Nimbbits para sistemas operativos Linux y construir una infraestructura propia, o por otro lado, se puede registrar a una nube privada y escalarla a cualquier tamaño.

Para este trabajo, se analizan las características de la Nube Pública de Nimbbits, la cual es una instancia del servidor de Nimbbits con funcionalidades limitadas que corre continuamente y no tiene ningún costo de uso. Nada más se necesita registrar una cuenta local en la plataforma o iniciar sesión con una cuenta de Google.

- **XAMPP para Windows 7.1.10**

Apache, MariaDB, PHP y otros componentes. Puede encontrar más información en la sección de Preguntas frecuentes o consultar las Guías de procedimientos para comenzar con las aplicaciones de PHP.

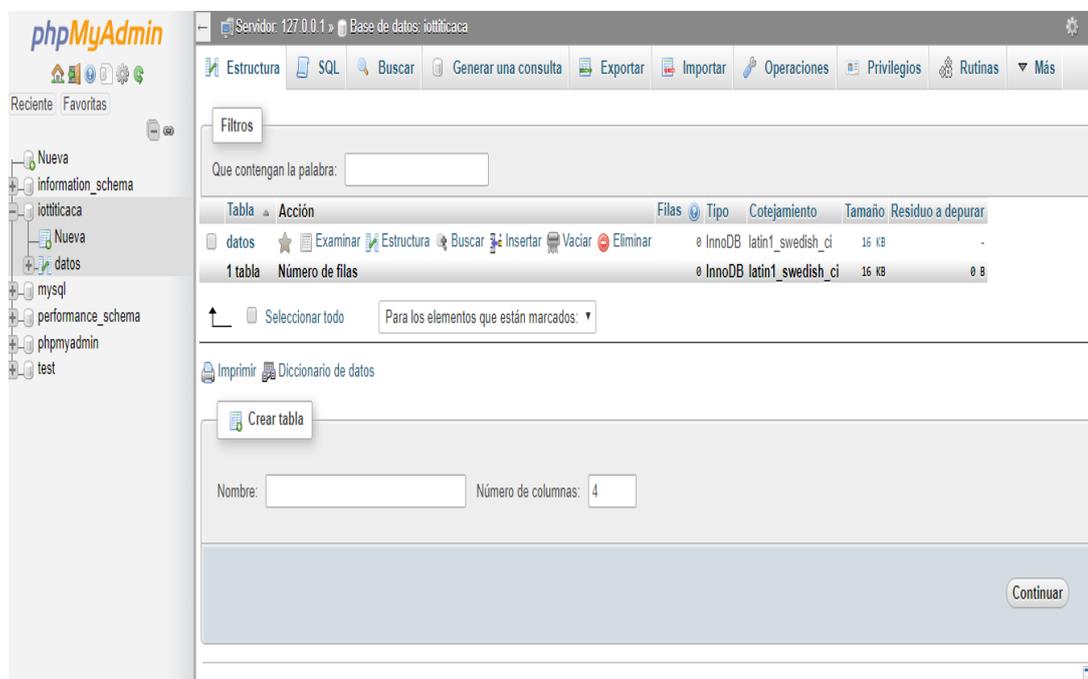


Figura N° 16 Entorno de desarrollo PhpMyadmin

Fuente: PhpMyadmin 7.1.10

XAMPP solo tiene fines de desarrollo. Tiene ciertas configuraciones que facilitan el desarrollo local pero que son inseguras si desea que otras personas puedan acceder a su instalación. Si desea que su XAMPP sea accesible desde Internet, asegúrese de comprender las implicaciones y verifique las preguntas frecuentes para saber cómo proteger su sitio. Alternativamente, puede usar WAMP, MAMP o LAMP, que son paquetes similares que son más adecuados para la producción.

2.6 HIPOTESIS

2.6.1 HIPÓTESIS GENERAL

La red FOG con internet de las cosas aplicadas en el monitoreo ambiental del Lago Titicaca mostrara los parámetros en tiempo real del estado de contaminación del Lago Titicaca.

2.6.2 HIPÓTESIS ESPECÍFICOS

- El diseño de red Fog con IoT del sistema de monitorización permitirá la transmisión de datos simulado en Cisco Packet Tracer.
- La implementación de un prototipo para la monitorización será capaz de transmitir información obtenida por los sensores y se adaptara a las condiciones ambientales del Lago Titicaca.
- Con el prototipo implementado se mostrara algunos parámetros ambientales en tiempo real del Lago Titicaca.

CAPITULO III

MATERIALES Y MÉTODOS

3.1 MATERIALES

3.1.1 HARDWARE

- Laptop Compaq hp presario CQ43
- Sensor de temperatura
- Sensor de PH
- Sensor de turbidez
- Nano arduino v3.1
- Módulo wifi RSR232-B
- Panel solar 7.2v
- Baterías ion-litio

3.1.2 SOFTWARE

- Sistema Operativo de 32 bits Windows 7
- Arduino Ide
- Sublime Text versión 3.0 Build 3143

- Gestor de base de datos mysql, sql.
- XAMPP control panel v3.2.2
- Software de simulación Packet Tracer v7.0
- Microsoft Excel Profesional Plus 2013 v15.0.4569.1506

3.2 MÉTODO

3.2.1 TIPO DE ESTUDIO

Es exploratoria, porque se examina un tema o problema de investigación poco estudiado. Es correlacional por que tiene como propósito medir el grado de relación entre las variables de las hipótesis planteadas; así como también es descriptiva, porque no se da la manipulación de variables, estas se observan y se describen tal como se presentan en la realidad, su metodología es fundamentalmente descriptiva, aunque puede valerse de algunos elementos cuantitativos y cualitativos.

Población

La población de estudio es pequeña, ya que se utilizaron tres nodos de transmisión, se ha considerado todo el universo como la población.



Figura N° 17 Bahía del Lago Titicaca

Fuente: Google Earth pro

3.2.2 TÉCNICAS DE PROCESAMIENTO Y ANÁLISIS

Plan de recolección de datos:

Los datos obtenidos sobre el monitoreo de sensores, se realizará según la fase y el software o equipo a utilizar. Se siguieron los siguientes pasos para la Recolección de datos:

- Se armó la topología del nodo de sensor de prueba con equipos reales y sensores.
- Se realizó la programación en c#, c, socket y más programas de interacción con los sensores para que los datos muestreados sean correctos y visualizados en tiempo real.

Los datos de campo obtenidos en referencia a la determinación del tiempo de llegada y tasa de transferencia de bits en los tres casos en la topología dada van a ser analizados e interpretados a través de las herramientas informáticas adecuadas.

a) DISEÑO DE UNA RED FOG BASADO EN INTERNET DE LAS COSAS

Para el diseño de una red Fog basado en internet de las cosas se consideró el protocolo de comunicación MQTT, Usando redes de sensores inalámbricos y otro según el enfoque utilizado en Internet de la cosas (IoT). En ambos casos utilizaremos plataformas Arduino que maneja la recolección de los datos provenientes de sensores que se anexan a las mismas.

Para la transmisión de los datos se utiliza el protocolo MQ Telemetry Transport (MQTT) dado que se encuentra disponible para plataformas Arduino y puede ser utilizado en ambos abordajes. Funciona sobre el protocolo TCP / IP está diseñado para conexiones con ubicaciones remotas donde se requiere una "huella de código pequeña" o el ancho de banda de la red es limitado. El patrón de publicación publicación-suscripción requiere un intermediario de mensajes, con la cual perfectamente se integra el propósito de investigación. Para desarrollar este propósito se utilizó el software Cisco Packet Tracer que dispone una serie de herramientas de diseño y simulación de Redes de IoT .

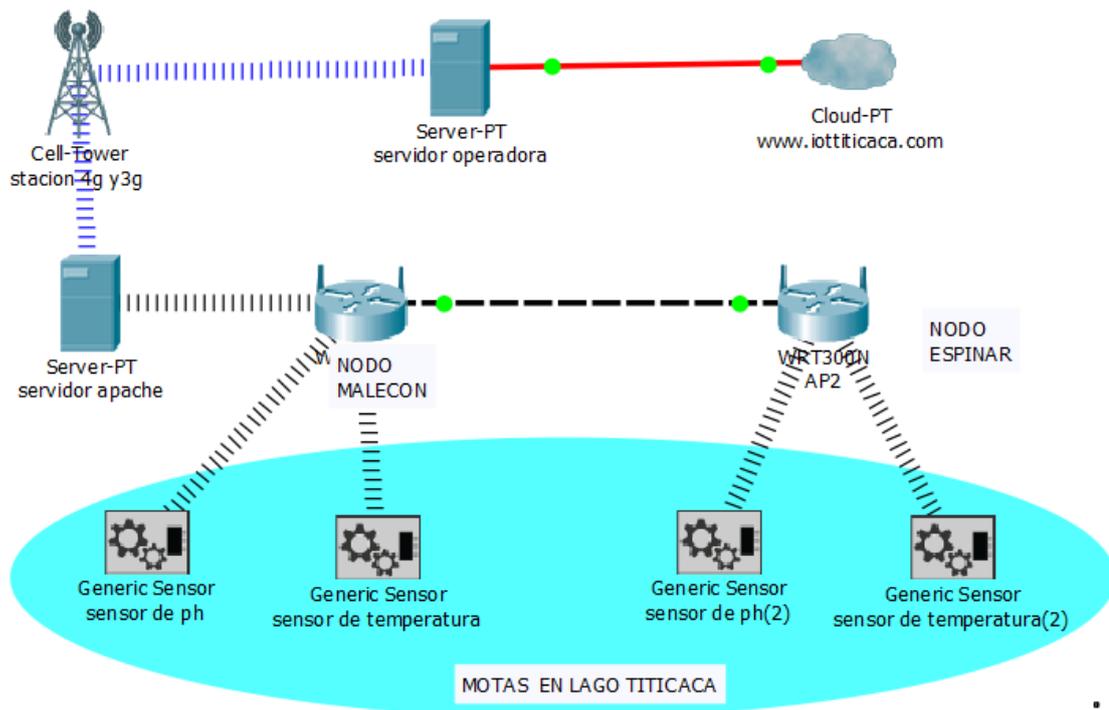


Figura N° 18: Diagrama simulado de Red FOG en Cisco Packet Tracer

Elaboración: Propia

En la Figura N° 18 se observa la red diseñada en Cisco Packet Tracer que consta de sensores que estarán ubicados en la superficie del Lago Titicaca, y luego transmite a través de Wi-Fi de 2.4Ghz hasta el nodo en donde se conecta con un Router para luego hacer un enlace hacia el servidor.

Diseño de enlace de la red con radio Mobile

Para los enlaces de larga distancia de Nodo 1 (malecón), Nodo2 (Espinar), y el servidor ubicado en la EPIE- UNAP se simuló en Radio Mobile que es un programa de simulación de radio enlace gratuito que nos sirve para operar dentro del rango de 20 MHz a 20 GHz, basado en el modelo de propagación ITS (Irregular Terrain Model).

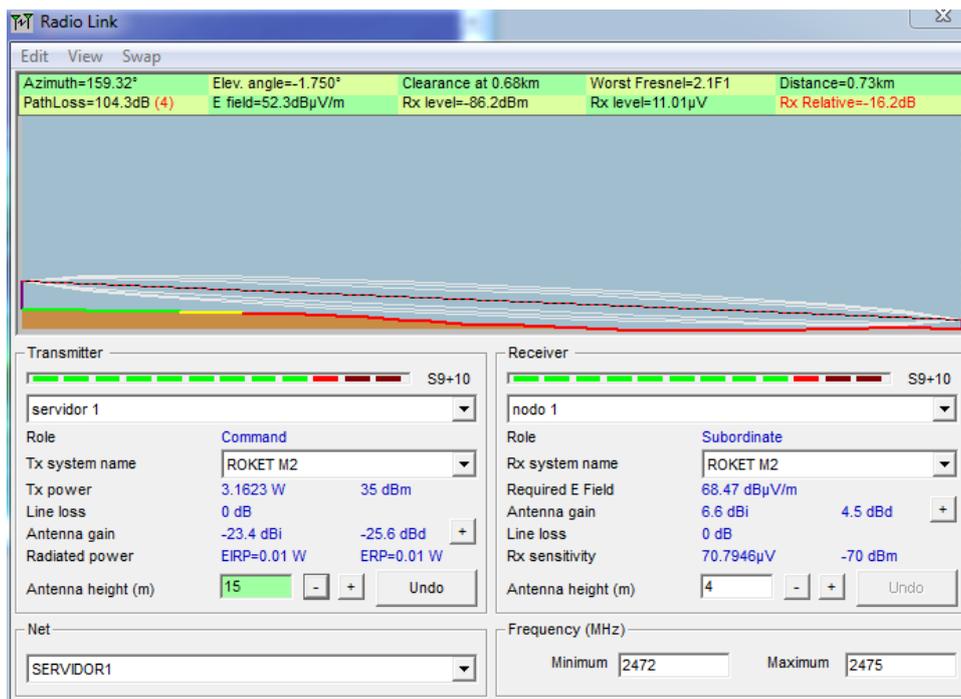


Figura N° 19: Calculo de enlace con Radio Mobile Nodo 1 y servidor EPIE

Elaboración: Propia

En la figura 19 se puede observar el enlace diseñado en Radio Mobile desde el servidor hasta el primer nodo que en la tabla se detalla las características.

Tabla N° 7 : Características de enlace nodo 1 y servidor

Punto servidor	Nodo 1	
latitud	-15.82428	-15.83042
longitud	-70.01785	-70.01544
Potencia TX	35 dBm	
Potencia RX	-70bBm	
Ganancia de antena	-23.4dBi	
Distancia total de enlace	731 m	

Elaboración: Propia

Para el otro enlace a considerarse en el diseño de la Red es desde el punto Espinar hacia el Nodo 1 el cual será un punto de conexión hacia el servidor.

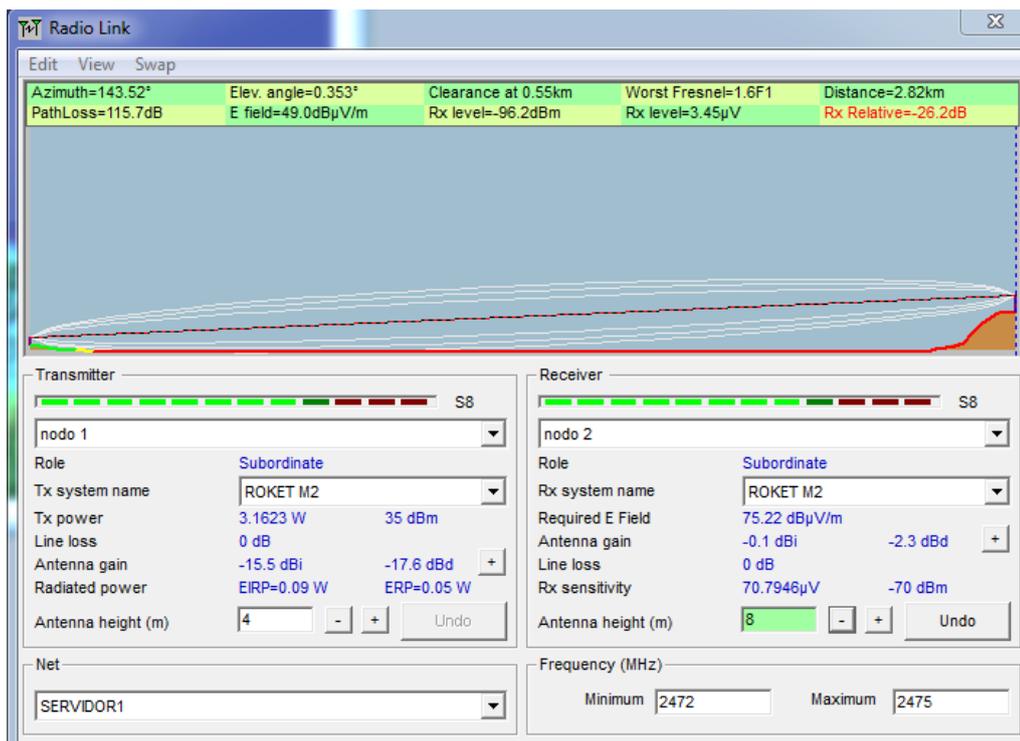


Figura N° 20: Calculo de enlace con Radio Mobile Nodo 1 y Nodo 2

Elaboración: Propia

En la figura N° 20 se considera una conexión de larga distancia desde en nodo Espinar y el nodo 1 ubicada en el malecón siendo detallada las características de conexión en el cuadro siguiente.

Tabla N° 8: Características de enlace nodo 1 y servidor

Nodo 1	Nodo 2	
Latitud	-15.83042	-15.83041
Longitud	-70.01544	-69.99973
Potencia TX	35 dBm,	
Potencia RX	-70bBm	
Ganancia de antena	-2.3dBi	
Distancia total de enlace	2083 m	

Elaboración: Propia

b) DISEÑO DEL PROTOTIPO DE UN NODO

Un nodo está constituido por varios motas los nodos interactúan con las motas que están constituidas por sensores ,Gateway y hardware, Tomando en cuenta esa descripción se diseñó un prototipo puesto que en el mercado no existen equipos con esas características disponibles en nuestro medio aplicables al ambiente adverso del Lago Titicaca.

- **Diseño de un prototipo de mota**

Está constituido por procesadores o placas de comunicación por radio, sensores y hardware para la adquisición de datos tal como lo describe Samaniego Armijos. Como se muestra en la figura 21, para poder gestionar los datos registrados por los sensores se utilizó un nano arduino como procesador y estandarización de datos analógicos y algunos digital, para luego ser enviado a través de una conexión wifi el cual tiene su respectivo dirección Ip y Mac dichas características está integrado en la tarjeta de comunicación WifiR232, para poder conectarse a un nodo de red y finalmente enviado hacia el servidor. La estructura se diseña de acuerdo al requerimiento necesario que pueda ser resistente al calor frio y la humedad además con alimentación propia solar.

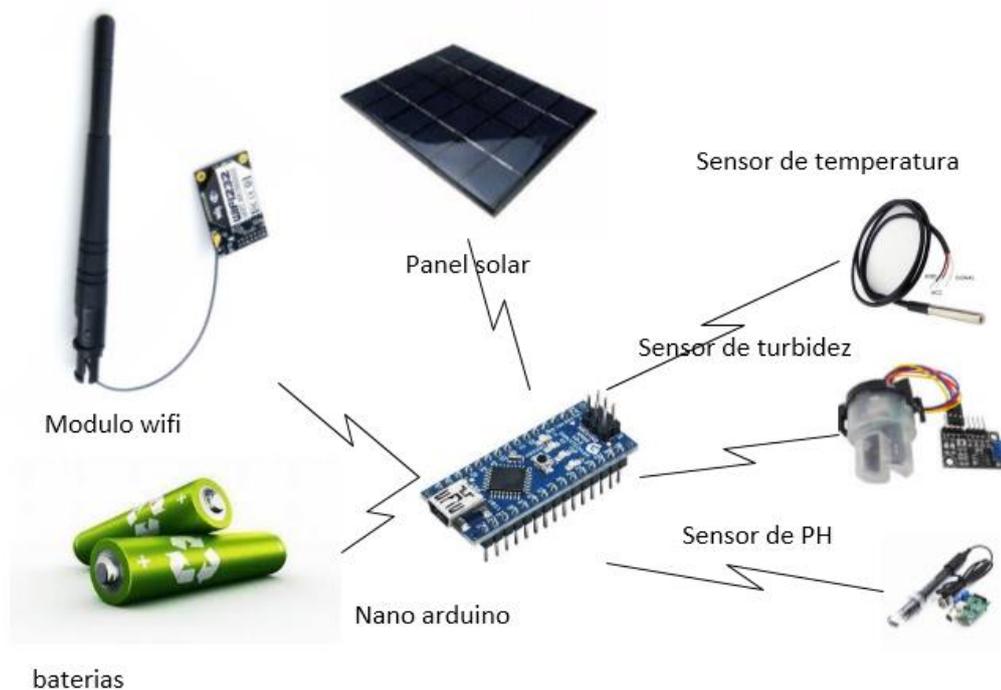


Figura N° 21: Esquema de prototipo de WSN

Elaboración: Propia

Consumo de energía

El consumo de energía está calculado de acuerdo al requerimiento promedio de todos los dispositivos instalados en una mota se calcula para alimentar de suficiente energía para cada WSN, y el consumo total se detalla, el consumo promedio es por el tiempo de transmisión, es muy determinante el tipo de batería y fuente de carga que contiene el dispositivo diseñado.

Nano arduino	19 mA
Sensores	30mA
Módulo wifi232	300mA
Total	349 mA

Para hacer cálculo de consumo de energía por hora se aplica la fórmula de conversión a watts. Multiplica los amperios por los voltios.

$$0.349 \text{ Amp} \times 5\text{V} = 1.745 \text{ Watts}$$

Multiplica por las horas de uso al día. El vatiaje te dice el ritmo al que se consume la energía mientras el dispositivo está activo. Multiplica el vatiaje por el número de horas por las que funciona el dispositivo durante un día promedio.

$$1.745\text{watt} \times 0.5 \text{ horas/día} = 0.8725\text{w/hora}$$

Divide entre 1.000. Esto convierte los vatios-hora en kilovatios-hora.

$$0.8725 \text{ vatios hora/día entre } 1000 \text{ vatios/Kilovatio} = 0.00087 \text{ kw-hora al día.}$$

Para ese propósito del funcionamiento de la mota se requiere un mínimo consumo de energía.

CALCULO DE ALCANCE DE CONEXION

$$\text{Answer: } 19.5 \text{ dBm} = -10.5 \text{ dBW} = 89.1251 \text{ mW} = 0.0891 \text{ W}$$

Cómo convertir entre (dBW, dBm, W y mW):

$$\text{mW} = 10(\text{dBm}/10) = 10(19.5/10) = 89.1251 \text{ mW}$$

$$\text{W} = 10(\text{dBW}/10) = 10(-10.5/10) = 0.0891 \text{ W}$$

$$\text{W} = \text{mW}/1000 = 89.1251/1000 = 0.0891 \text{ W}$$

$$\text{mW} = \text{W} \times 1000 = 0.0891 \times 1000 = 89.1251 \text{ mW}$$

De acuerdo al cálculo realizado en la práctica la cobertura de alcance es de unos 400 a 500 metros de acuerdo a las condiciones ambientales del Lago

Titicaca. Se debe de considerar las condiciones ambientales como el oleaje, lluvia en general la inestabilidad de las aguas los cuales afectan la transmisión.

c) DISEÑO Y CREACIÓN DE UNA PLATAFORMA VIRTUAL

- Creación de server en phpMyAdmin

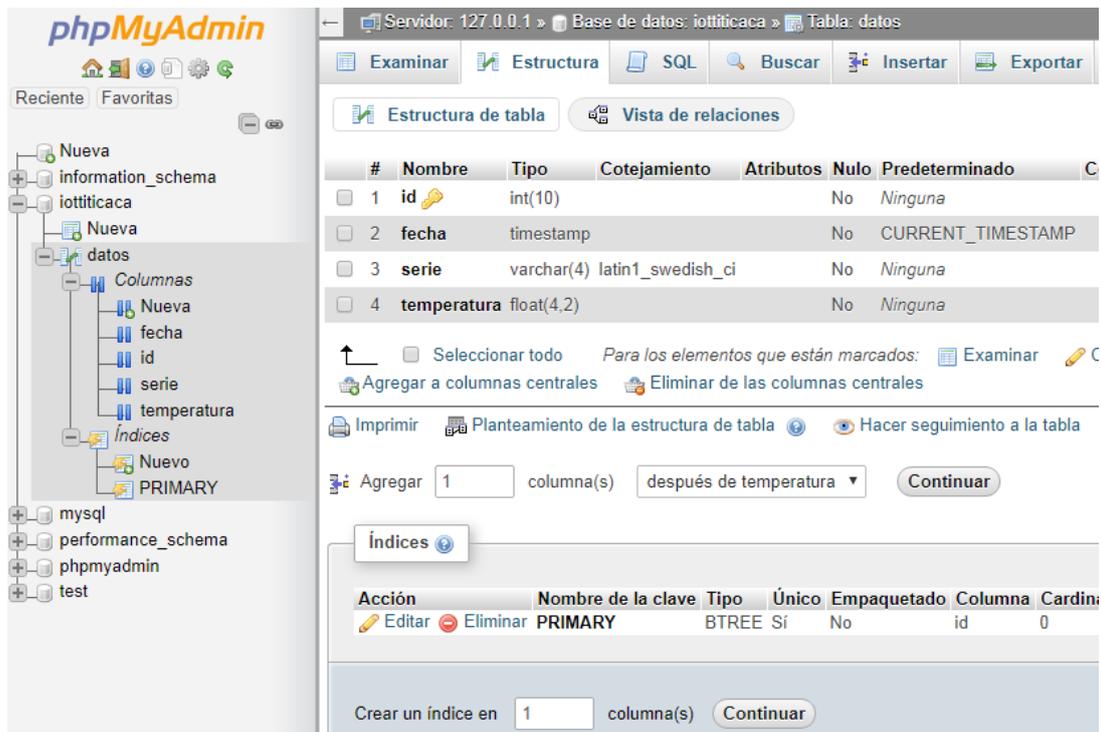


Figura N° 22: Plataforma de creación del servidor.

Fuente: Entorno de desarrollo phpMyadmin

- Portada de inicio de monitorización

La portada de bienvenida de la plataforma de monitorización de redes de sensores se enfoca en la facilidad de uso y en el acceso a los usuarios de manera que sea una interfaz intuitiva. Los navegadores que soporten HTML5 como Chrome, Firefox y Opera son altamente recomendados tanto para computadores, tabletas y teléfonos móviles.

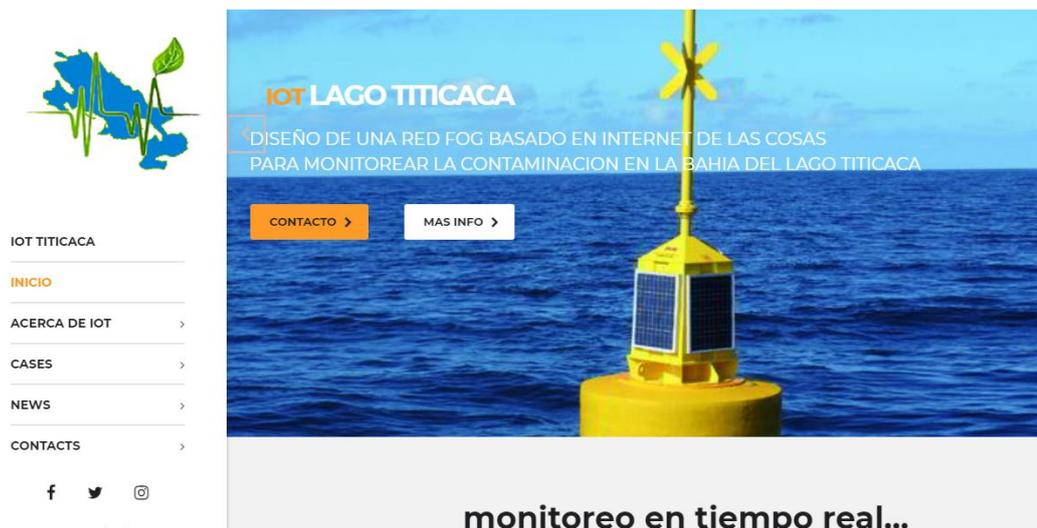


Figura N° 23: Portal web de IoT Titicaca

Elaboración: Propia

- **Registro de cuenta**

La interfaz de registro de una cuenta permite al usuario inscribirse con sus datos personales, ingresando sus nombres y apellidos, correo electrónico (de cualquier dominio), contraseña y el motivo de uso. Los campos requeridos son validados tanto en longitud como el uso de caracteres permitidos y que puedan registrarse una vez.

- **Gestión del entorno de monitorización**

Los invitados o personas no registradas en la plataforma, también pueden conocer el comportamiento de los sensores disponibles de forma limitada bajo una interfaz de listado cronológicamente ordenado y con las opciones de filtrados según el tipo de sensor, nodo o fecha registrados.

Por otro lado, a partir de la activación de la cuenta del usuario, se establece su perfil y podrá emplear las funcionalidades de la plataforma según sus intereses.



IOT TITICACA

INICIO

ACERCA DE IOT

CONTACTO

REGISTRARSE

INGRESAR

IOT



UNIVERSIDAD NACIONAL DEL ALTIPLANO - PUNO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



MONITOREO IOT TITICACA

HUMEDAD HISTORIAL

Humedad relativa sobre la superficie del Lago Titicaca



TEMPERATURA HISTORIAL

Sensor de temperatura superficial del Lago Titicaca

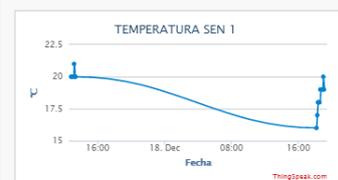


Figura N° 24: Portal de monitoreo

Elaboración: Propia

El usuario puede apreciar un entorno en tiempo real, con un período de actualización de acuerdo con la recolección de la data, el comportamiento de los sensores con sus respectivos estándares y los estados de las alertas encontradas.

Para obtener la data actualizada se colocó un script que verifica si ha llegado una trama al nodo sensor con una frecuencia determinada por el especialista o administrador.

- **Historial**

En el módulo de Historial se proporciona las mismas funcionalidades mencionadas anteriormente, es decir, la sección multiparámetro, la comparación con estándares, las alertas y la exportación de gráficas. A éstas, se agregaron las funcionales de selección de un rango o período de interés, y la comparación entre diferentes sensores por área o diferentes áreas por sensor, y también se

puede visualizar el dato máximo, mínimo, o la moda del sensor seleccionado. Como se muestra en la figura N°25.

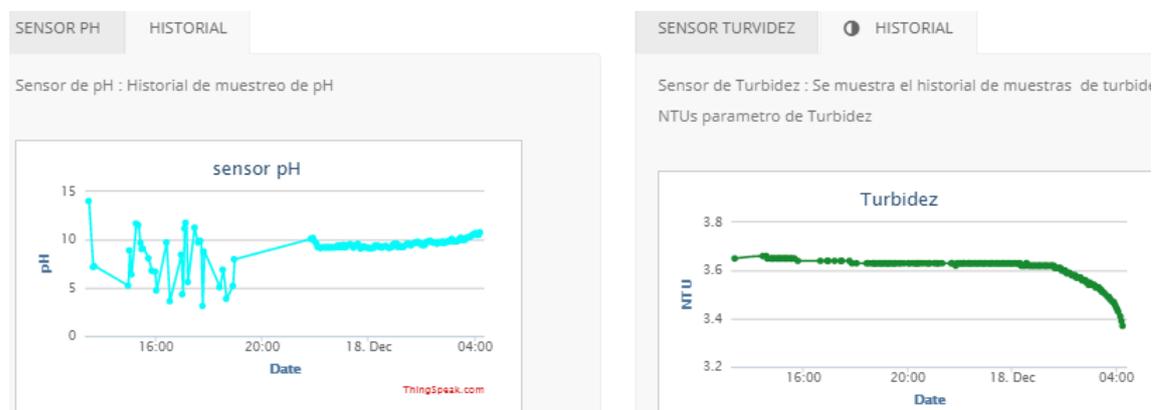


Figura N° 25: Historial de datos

Elaboración: Propia

- **Registro de sensores, nodos y parámetros**

Para el registro de sensores se tiene que completar los campos obligatorios de nombre de sensor, unidad de medición y tiempo de muestro. Así como también, los nodos sensores al registrarlos son necesarios los campos de ubicación, dirección IP y MAC.

Los parámetros, que son la asociación de un nodo sensor con sus respectivos sensores, se registran según la disponibilidad de los nodos y sensores que se hayan inscrito previamente. Se incluye la actualización de sus características, que el usuario con privilegios de especialista puede realizar.

Field 1 Chart Options

Title:	TEMPERATURA SEN 1	Timescale:	
X-Axis:	hora	Average:	
Y-Axis:	temperatura	Median:	
Color:	#0582CE	Sum:	
Background:	#ffffff	Rounding:	
Type:	spline	Data Min:	
Dynamic?:	true	Data Max:	
Days:		Y-Axis Min:	
Results:	20	Y-Axis Max:	

Save Cancel

Figura N° 26: Registro de sensores

Elaboración: Propia

- Generación de reportes

Otra funcionalidad más resaltante de la plataforma es la creación de reportes.

A través de ésta es posible la selección de data que sea de interés y será mostrada según el período de análisis, los sensores, los nodos, las alertas o los estándares que convenga.

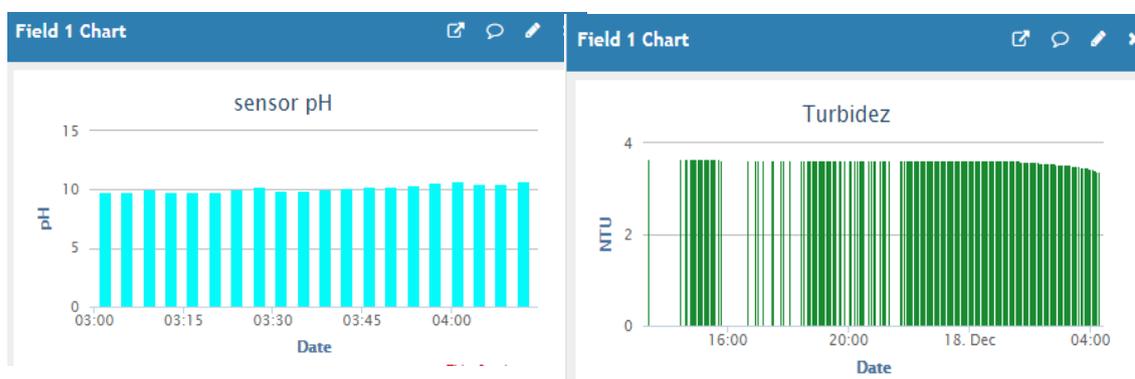


Figura N° 27: Generación de Reporte.

Elaboración: Propia

Antes de la generación del reporte, se validará si se dispone de la data requerida y se podrá hacer un reporte. También se podrá realizar el reporte por fechas de registros.

La generación de reportes es en forma automática en la sección de historial de reportes donde se muestra un periodo de los últimos 30 días de reporte, siendo esta modificable de acuerdo al requerimiento del usuario.

3.2.3 INSTRUMENTO DE RECOLECCIÓN DE DATOS

Técnicas

La técnica para la recolección de datos es mediante la observación, ya que el investigador actúa sobre los hechos con la ayuda de algún instrumento.

Instrumentos

Los instrumentos son la guía de observación del campo: esto indica los pasos que adopta el investigador para una buena estrategia en la observación de datos.

CAPITULO IV

RESULTADOS Y DISCUSIÓN

4.1 RESULTADOS DEL DISEÑO DE RED FOG CON INTERNET DE LAS COSAS

Los resultados del diseño de la Red Fog con IoT son simulados en software de comunicación como Radio Mobile y Cisco Packet Tracer los cuales pueden ser referenciales al momento de instalar los equipos reales en la implementación. EL protocolo de comunicación que se utilizo es el MQTT, y usando redes de sensores inalámbricos y otro según el enfoque utilizado en Internet de la cosas (IoT). En ambos casos utilizamos plataformas Arduino o compatibles que manejan la recolección de los datos provenientes de sensores que se anexan a las mismas. Para la transmisión de los datos se utiliza el protocolo MQ Telemetry Transport (MQTT) dado que se encuentra disponible para plataformas Arduino y puede ser utilizado en ambos abordajes. El protocolo MQTT es un protocolo de transporte de mensajes en la forma cliente / Servidor. Es liviano, abierto, sencillo y está diseñado para que sea fácil de implementar. Estas características lo hacen ideal para su uso en muchas situaciones, incluyendo entornos limitados como

para la comunicación de máquina a máquina (M2M) y el Internet de las Cosas (IoT), contextos en los que se requiere una pequeña huella de código y/o el ancho de banda es un bien escaso. El protocolo se ejecuta a través de TCP/IP, o sobre otros protocolos de red que proporcionan conexiones ordenadas, sin pérdidas y bidireccionales. Por otra parte se consideró la red de radio enlace de larga distancia por considerarse puntos críticos de donde se obtuvieron datos y ello se calculó de acuerdo a los parámetros de radio enlace utilizado software Radio Mobile.

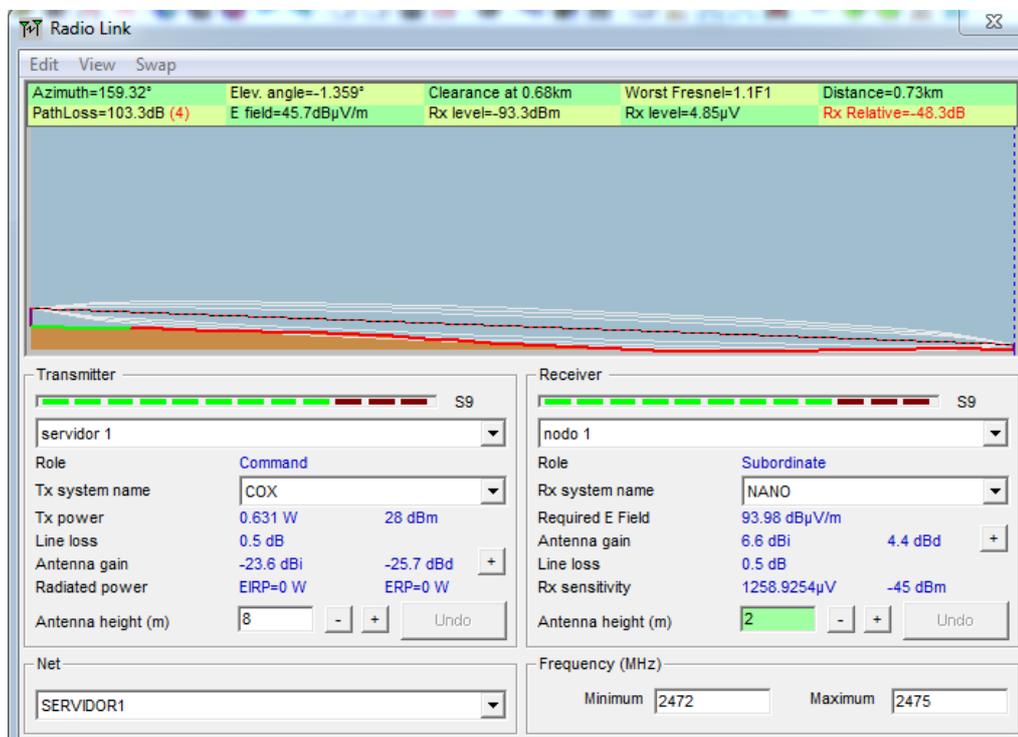


Figura N° 28 : Radioenlace del servidor hasta el nodo

Elaboración: Propia

En la figura se simula un radio enlace con software radio Mobile el primer enlace que se diseñó está considerado desde el servidor ubicado en la Escuela Profesional de Ingeniería Electrónica donde se ubicará el servidor hasta el

primer nodo ubicado en el malecón, de manera precisa se detalla en la tabla N°9 las características del nodo.

Después de hacer el radio enlace se considera el diseño de un nodo según el protocolo de comunicación de una red IoT, el nodo está diseñado con tres sensores independientes con comunicación wifi de corta distancia, el nodo está asignado con su respectivo dirección Ip y dirección Mac, los cuales se detallan en cuadro siguiente, se diseñó en el software Cisco Packet Tracer v.7.1.

Tabla N° 9: Direcciones Ip y Mac del nodo

	Nodo	Mota sensor1	Mota sensor 2	Mota sensor 4
Equipo de comunicación	Router Ap HUAWEI	Wifi 232	Wifi 232	Wifi232
Dirección IP	192.168.8.1	192.168.8.104	192.168.8.105	192.168.8.106
Dirección Mac	0C-5B-8F-27-9A-64	KC-5B-87-27-9A-64	RC-5B-8F-25-9A-64	HC-5B-8F-27-9A-64
Alcance en km		400m	400m	400m

Elaboración: Propia

El nodo del sensor está gestionado por un Router, que también puede ser un modem AP 3g o 4g que pueda hacer la gestión de conexión con el servidor, el Router gestiona los datos desde las motas para transmitir y/o conectarse con otros nodos cercanos y gestionar la mejor ruta para el envío de datos hacia el servidor.

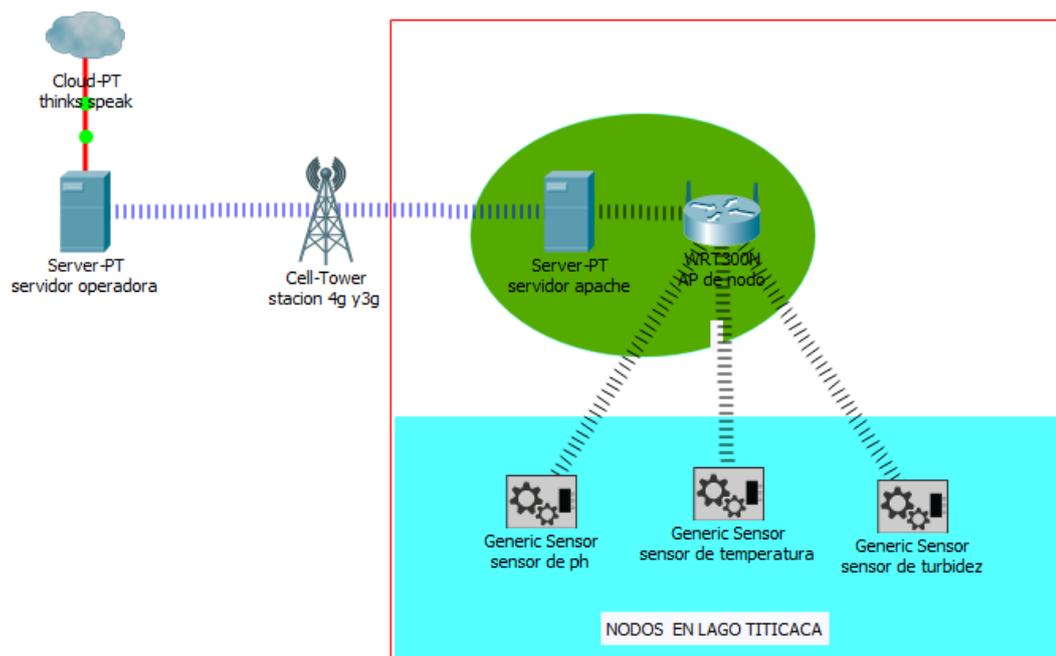


Figura N° 29 : Diseño de la red en Cisco Packet Tracer

Elaboración: Propia

4.2 RESULTADOS DEL DISEÑO DEL PROTOTIPO PARA EL MONITOREO AMBIENTAL.

Al no encontrar en el mercado dispositivos integrados basados en IoT para monitorizar parámetros ambientales en condiciones como el Lago Titicaca se consideró elaborar un prototipo basado en la arquitectura IoT y red de sensores. La investigación inicio con busca de información sobre plataformas y tarjetas de desarrollo que cumplieran con los requerimientos que precisan este tipo de mediciones en campo, se exploró en el campo de las tarjetas de desarrollo, Arduino, rasperry-pi, para dar con hardware apropiado, se exploró también las opciones existentes para transmisión de información desde el interior del Lago Titicaca lugar totalmente inestable para transmisión de datos.

Al final del estudio se eligió el prototipo de transmisión con las siguientes características:

Tabla N° 10: Descripción del prototipo de una Mota

DESCRIPCIÓN	UNId./MEDIDA
Consumo de energía	350 mA
Peso aprox.	600 gms
Capacidad de conexión	Wifi 2.4 Ghz
Máximo alcance de transmisión	500m
Costo aproximado	350 soles
Soporta conexión sensores	Analógicos y digitales

Elaboración: Propia

En la figura N° 30 se muestra un prototipo elaborado con dispositivos existentes en el mercado adecuando a las necesidades planteadas en el proyecto de investigación.

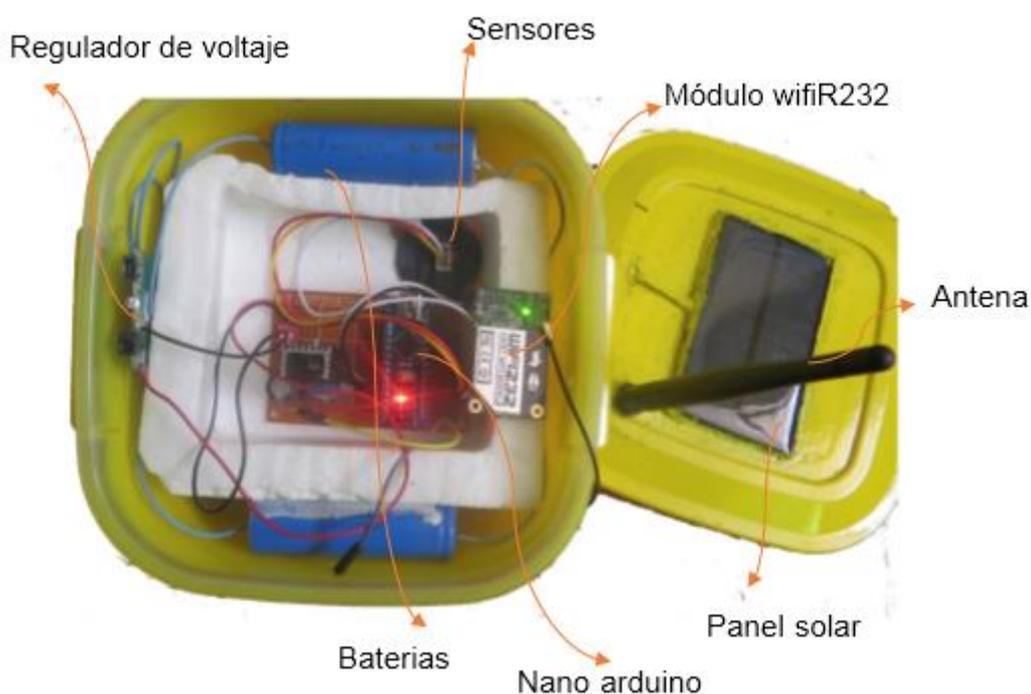


Figura N° 30: Prototipo de una Mota

Elaboración: Propia

4.3 RESULTADOS MONITORIZACIÓN DEL LAGO TITICACA

Los puntos críticos que se consideró son lugares donde desembocan aguas servidas en forma directa o tratada, que son conocidas por el público en general dichas zonas se consideraron como nodos de comunicación una de ellas considerado como NODO 2 ubicado en la desembocadura de la laguna de sedimentación espinar.

El siguiente punto a considerarse como NODO 1 fue el malecón eco turístico alrededor de la desembocadura de aguas servidas de la Universidad Nacional del Altiplano.

4.3.1 MUESTREO EN UN NODO

El proceso de envío de datos inicia desde que una mota se conecta hacia la red del nodo dicha conexión es gestionada por el Arduino mediante comandos AT que identifica la red pre programada o un punto de conexión disponible.

4.3.2 CONEXIÓN DEL NODO AL SERVIDOR

En este proceso se inicia cuando la tarjeta de red del sensor identifica a la red donde se conecta con el comando AT+CWJAP, posteriormente el comando AT+CIPSTART identifica el protocolo de conexión y la dirección Ip y el puerto del servidor y confirmando la conexión con el comando AT+CIPSEND y la respuesta del servidor confirmando con la clave de aceptación del nodo "0TNGECPR7MEDKWC4", Este último proceso se repite cada vez que hay un reporte.

```
AT
AT+CWMODE=3
AT+CWJAP="TITIKAKA","titicaca2016"
AT+CIPSTART="TCP","184.106.153.149",80
AT+CIPSEND=48
GET /update?key=0TINGECPR7MEDKWC4&field1=19.000
```

Figura N° 31: Proceso de conexión mediante comandos AT

Elaboración: Propia

4.3.3 ENVIO DE DATOS AL SERVIDOR

El envío de los datos realiza inicializando el módulo wifi reconociendo la red disponible o pre configurada haciendo ping con la Ip del servidor, luego el servidor responde verificando el Key o llave de conexión de esta manera se asegura los datos enviados hacia el servidor, este proceso es repetido cada vez que se envía los datos.

```
AT
AT+CWMODE=3
AT+CWJAP="TITIKAKA","titicaca2016"
AT+CIPSTART="TCP","184.106.153.149",80
AT+CIPSEND=48
GET /update?key=0TINGECPR7MEDKWC4&field1=16.000
AT+CIPSTART="TCP","184.106.153.149",80
AT+CIPSEND=47
GET /update?key=0TINGECPR7MEDKWC4&field2=44.00
AT+CIPSTART="TCP","184.106.153.149",80
AT+CIPSEND=48
GET /update?key=0TINGECPR7MEDKWC4&field1=17.000
```

Figura N° 32: Proceso de inicio de conexión al servidor

Elaboración: Propia

Los datos obtenidos por los sensores se envían con valores numéricos interpretados según la formula interpretada con la programación de Arduino como se muestra en la figura N°32, según el sensor utilizado luego es enviado hacia el servidor.

El proceso de envío se muestra en forma gráfica en la figura 35 donde se muestra la ruta completa del envío de datos iniciando desde el sensor hasta el almacenamiento en la nube.



Figura N° 33: captura de pantalla monitor serial Arduino.

Elaboración: Propia

Una vez enviado el servidor se encarga de almacenar datos según la disponibilidad de espacio contratado, los datos son mostrados en el portal en forma gráfica.

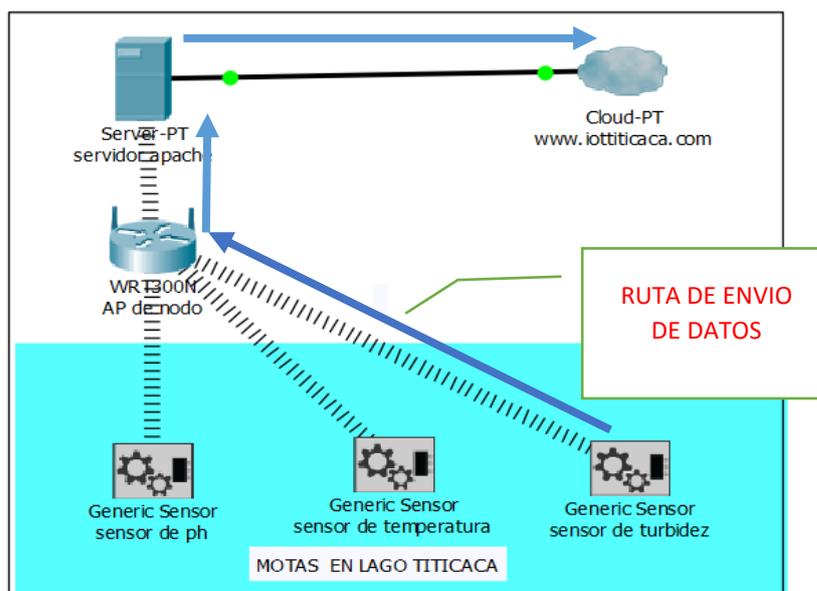


Figura N° 34: Diagrama de envío de Datos

Elaboración: Propia

Finalmente se muestra en el portal web los resultados que se procesan en el servidor que son accesibles para el público estos datos se pueden observar desde cualquier dispositivo con conexión a internet.



Figura N° 35 : Graficas en el portal www.iottiticaca.com

Elaboración: Propia

En la figura 35 se aprecia el monitoreo ambiental en tiempo real registrados en día estas muestras se reflejan a través de una gráfica diseñados para el entendimiento del público Usuario.

El portal web muestra los resultados que se procesan en el servidor que son accesibles para el público estos datos se pueden observar desde cualquier dispositivo con conexión a internet.

4.3.4 COMPARACION Y VALIDACION DE REPORTE

Para poder validar los datos se tomaron las fuentes de instituciones que se dedican al monitoreo ambiental del Lago Titicaca Uno de ellos es el Proyecto

Especial Binacional Lago Titicaca que tiene reportes en los puntos que se muestran en la figura dichos datos son reportados mensualmente.

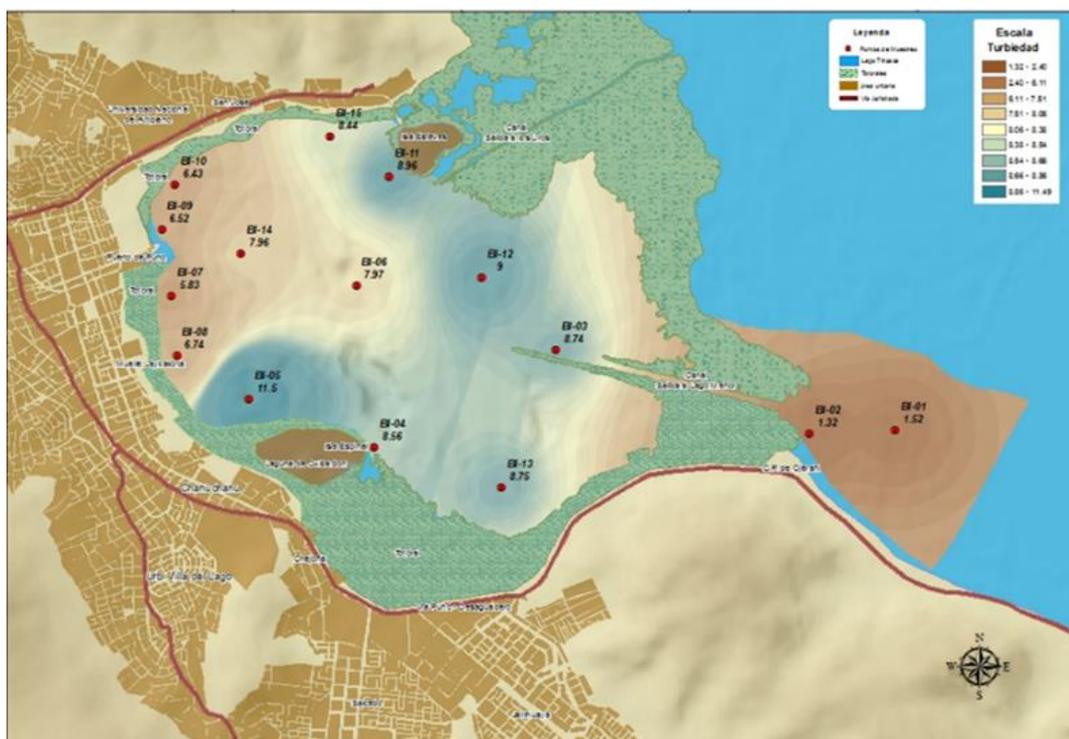


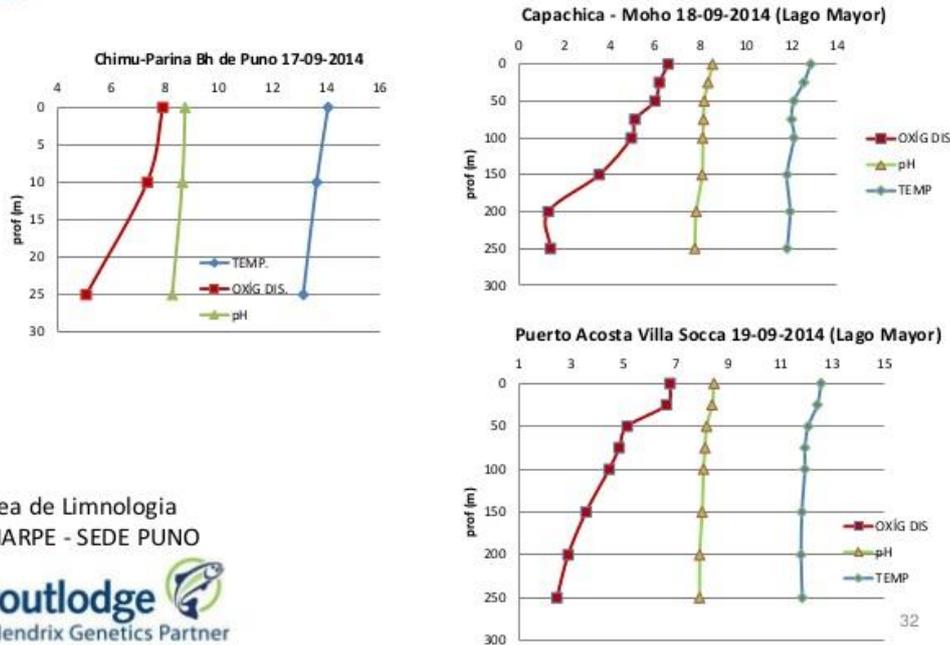
Figura N° 36: Reporte grafico de pH en la bahía del Lago Titicaca

Fuente: PEBLT octubre del 2017

Por otro lado El Instituto del Mar del Perú - IMARPE Organismo Técnico Especializado del Ministerio de la Producción, orientado a la investigación científica, así como al estudio y conocimiento del mar peruano y sus recursos, tiene un reporte que tiene mucha similitud en sus reportes al proporcionado por el PEBLT, Proyecto Especial Binacional Lago Titicaca.



Perfiles de temperatura, pH y oxígeno en columna de agua en Septiembre 2014



Area de Limnología
IMARPE - SEDE PUNO
Troutlodge
A Hendrix Genetics Partner

Figura N° 37: Reporte de IMARPE-SEDE PUNO

Fuente: IMARPE -Puno

Según los reportes de IMARPE Sede Puno se tiene una referencia de los análisis de pH y temperatura las cuales coinciden con los reportes obtenidos en la actualidad que a continuación se hace una comparación referencial.

Para hacer la validación se compara con los valores antes mostrados de dos instituciones dedicadas al estudio de las aguas del Titicaca para lo cual se muestran datos adquiridos por los sensores que fueron puestas en el lago Titicaca.

Por otra parte se debe de considerar la escala de valores de pH coeficiente que indica el grado de acidez o basicidad de una solución acuosa, que son considerados de valores desde 0 a 14 los valores de 0 a 7 son considerados acidas, el 7 se considera neutro y los valores de 7 a 14 como alcalinas.

```
pH= 9.50
pH= 9.50
pH= 9.54
pH= 9.52
pH= 9.53
pH= 9.52
pH= 9.52
pH= 9.51
pH= 9.51
pH= 9.50
pH= 9.52
pH= 9.52
pH= 9.52
pH= 9.52
pH= 9.52
```

Figura N° 38: Captura de pantalla monitor serial valores pH

Elaboración: Propia

En la figura 38 se observa un reporte de unos 15 minutos de transmisión de datos de pH con una oscilación de 9.50 a 9.54 el cual indica que las aguas del lago son alcalinas según la tabla de definición de la escala de pH, con intervalo de reporte de 2 minutos. Este reporte es capturado desde el monitor serial transmitida desde una mota hacia el servidor.

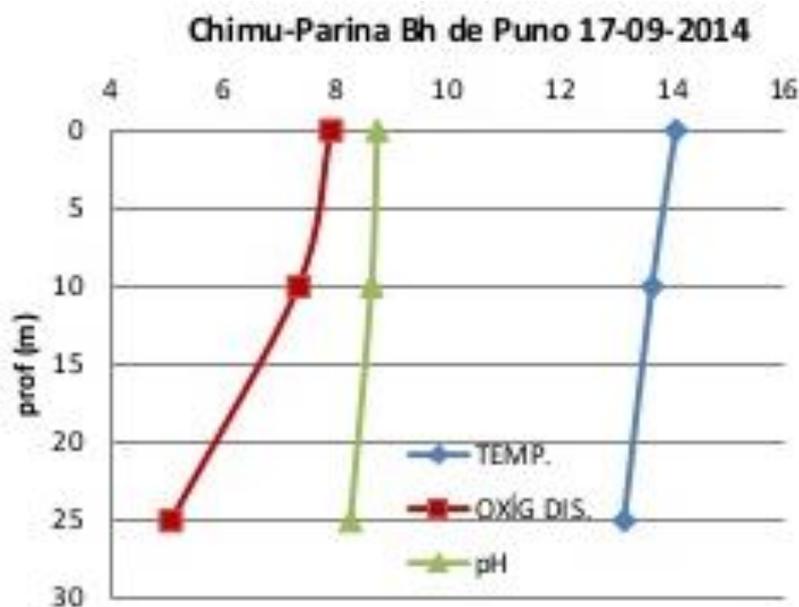


Figura N° 39: Reporte IMARPE-SEDE PUNO bahía de Puno

Fuente: IMARPE 2014

En la figura N°39 se observa la temperatura con la línea azul que tiene una tendencia desde los 12 a 14 grados, y pH en una variación de 7 a 9 los cuales son muy cercanos a los valores obtenidos con las motas colocadas en el Lago Titicaca, por otra parte se tomó muestras con un pH metro comercial el cual tiene una lectura de 9.88 tal como se muestra en la figura, este muestreo se realizó en la bahía del Lago Titicaca.



Figura N° 40: Medición de pH de las aguas del Lago Titicaca

Elaboración: Propia

De igual manera se hace una comparación de datos obtenidos por el sensor de turbidez de esta investigación con los datos reportados por el PEBLT.

Los valores obtenidos están en la escala de NTU, Unidades Nefelométricas de turbidez que mide la intensidad de la luz dispersada a 90 grados, El agua transparente a la vista del ojo humano es aproximadamente menores a 3 NTU de ahí se puede considerar la turbidez del agua.

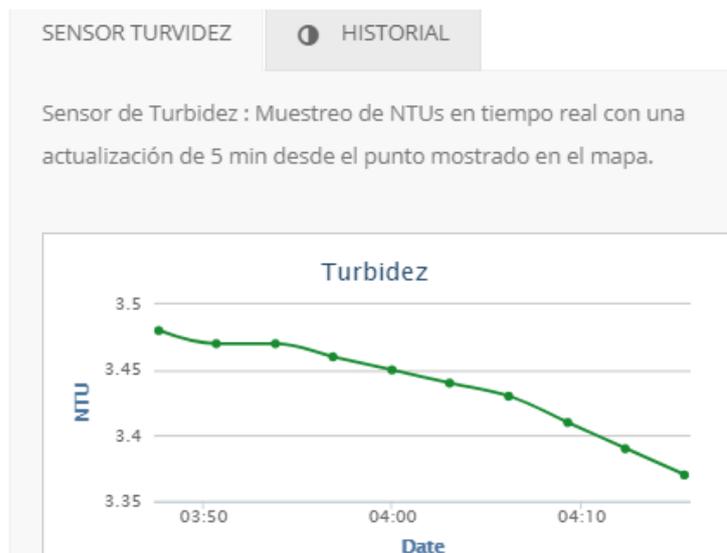


Figura N° 41: Reporte de sensor de turbidez

Elaboración: Propia

Los datos mostrados en la figura fueron obtenidos en un tiempo de 20 min con reportes de muestra de cada 3 min, dichas muestras pueden ser comparados por el reporte del Proyecto Especial Binacional de Lago Titicaca que son mostrados en forma gráfica.

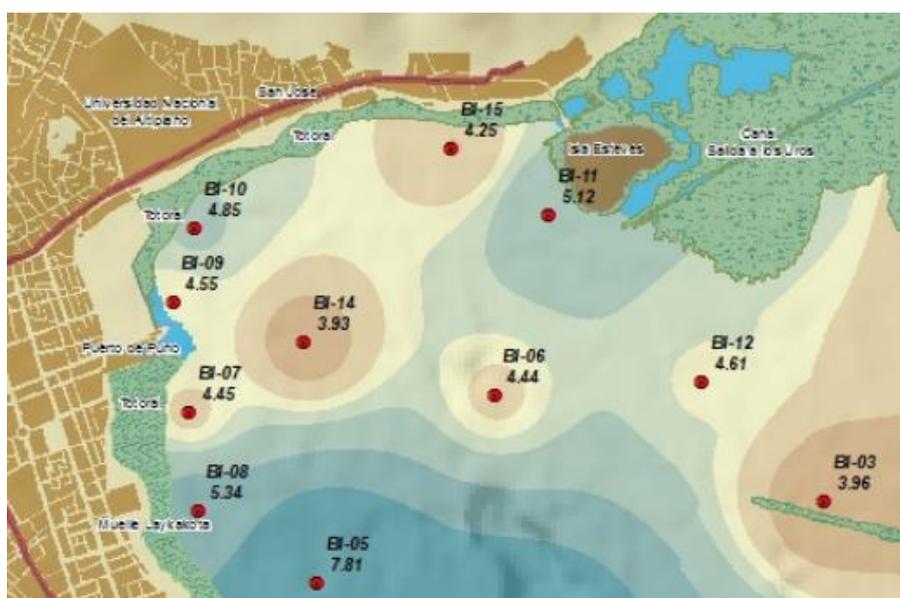


Figura N° 42: Reporte de turbidez en diferentes puntos

Fuente: PEBLT octubre del 2017

CONCLUSIONES

PRIMERO: El proceso de diseño de la red FOG con internet de las cosas se realizó simulando en software diseñados para elaborar una red de comunicaciones que son basadas en los protocolos de comunicación de IoT.

SEGUNDO: Para el diseño de la red de una red FOG basado en internet de las cosas se utilizó software de simulación para desarrollar una red con características basadas en IoT que tiene características particulares, como la asignación de direcciones IP, conexiones inalámbricas con resultados óptimos.

TERCERO: Se diseñó un prototipo básico con independencia energética y capacidad de transmisión de datos hacia un servidor, para monitorear condiciones ambientales de las aguas del Lago Titicaca que funcionan con protocolos de comunicación IoT, y costos muy accesibles para realizar mantenimiento, reposición en caso de daños avería, y se adaptan a las condiciones ambientales del Lago Titicaca.

CUARTO: Al implantar motas en el Lago Titicaca visualiza y procesa en tiempo real los datos recolectados de los sensores de temperatura, y pH del agua, son enviados hacia el servidor y se puede apreciar los resultados ingresando al portal web “www.iottiticaca.com” en donde se puede observar los resultados de parámetros ambientales con actualización en un intervalo de tiempo programable, además se puede apreciar el historial de sucesos. Estos valores pueden tener valores referenciales de las condiciones ambientales puesto que tienen un muestreo de un área muy pequeña las Motas de sensores.

RECOMENDACIONES

PRIMERO. Un aspecto fundamental del Internet de las Cosas es la relación cercana que tiene con las plataformas de Computación en la Nube (Cloud Computing). Las aplicaciones IoT están en realidad basadas en Computación en la Nube, de manera que sea posible proveer características de almacenamiento con mejor escalabilidad y gran interoperabilidad a través de accesos abiertos e interfaces directas para la comunicación e intercambio de información. El mejor modo de comunicar las aplicaciones propias con infraestructuras de nube, es mediante servicios web; específicamente, servicios web, los cuales son muy sencillos y livianos de usar. Por ello resultan ser los más adecuados para micro controladores y dispositivos que tengan un costo bajo.

SEGUNDO. Para poder monitorizar la contaminación ambiental se requiere un estudio de una red más amplia de sensores que tengan una cobertura mayor de área de reporte, y se debe tomar en cuenta algunos aspectos como calibrar los sensores de acuerdo a las especificaciones del fabricante a fin de obtener resultados óptimos en la investigación, utilizar otros sensores como detectores de sulfuros u otro parecido para tener más reportes de condiciones ambientales del Lago Titicaca.

TERCERO. Para tener reportes de contaminación ambiental en el Lago Titicaca se recomienda un estudio con profesionales medioambientalistas, hacer comparaciones de los resultados obtenidos con instituciones medioambientales certificadas, a fin de obtener resultados óptimos y confiables para la información del público interesado.

REFERENCIAS

- Arduino. (Julio de 2015). *Introduction to Arduino*. Obtenido de Introduction to Arduino: <http://www.arduino.cc/>
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A. (20 de Febrero de 2014). *Electrical Engineering and Computer Sciences*. Obtenido de Electrical Engineering and Computer Sciences.: <http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf>
- Armijos, S. (2015). *Red de Sensores HS*. Cuenca: Seminario Cieela.
- Beltrán, A. B. (2016). Arquitectura de referencia en el Internet of things. *SUNQU*, 14-15.
- Carballar, J. A. (2008). *VoIP. La telefonía de Internet*. Madrid, Magallanes, ESPAÑA: International Thomson Editores Spain Paraninfo, S.A.
- Chiang, M. (26 de Junio de 2015). *Fog Networks and the Internet of Things*. Obtenido de Fog Networks and the Internet of Things: <https://www.coursera.org/course/fog>
- Cisco. (2015). *Implementing Cisco Voice Communications and QoS (Vol. 1)*. Cisco.
- Corletti Estrada, A. (2016). *Seguridad en Redes*. Madrid, Moralarzal, ESPAÑA: AUTOR EDITOR.
- Doukas, c. (2012). *Building Internet of things With the Arduino*.
- EDUCAUSE. (27 de febrero de 2017). *EDUCAUSE Library CLOUD COMPUTING*. Obtenido de EDUCAUSE Library CLOUD COMPUTING: <http://www.educause.edu/library/cloud-computing>
- Electronics., M. (2014). *Desarrolle Aplicaciones para Internet of Things*. Reino Unido: MC Electronics.
- Hardy, Q. (24 de febrero de 2016). *The Era of Cloud Computing*. Obtenido de (T. N. Times, Editor): http://bits.blogs.nytimes.com/2014/06/11/the-era-of-cloud-computing/?_r=2
- Institute, Alexandra. (2011). Inspiring the Internet of Things. *IoT Comic Book*, 27.
- Instruments, N. (JULIO de 2017). *¿Qué es una Red Inalámbrica de Sensores?* Obtenido de ¿Qué es una Red Inalámbrica de Sensores?: <http://www.ni.com/wsn/whatis/esa/>

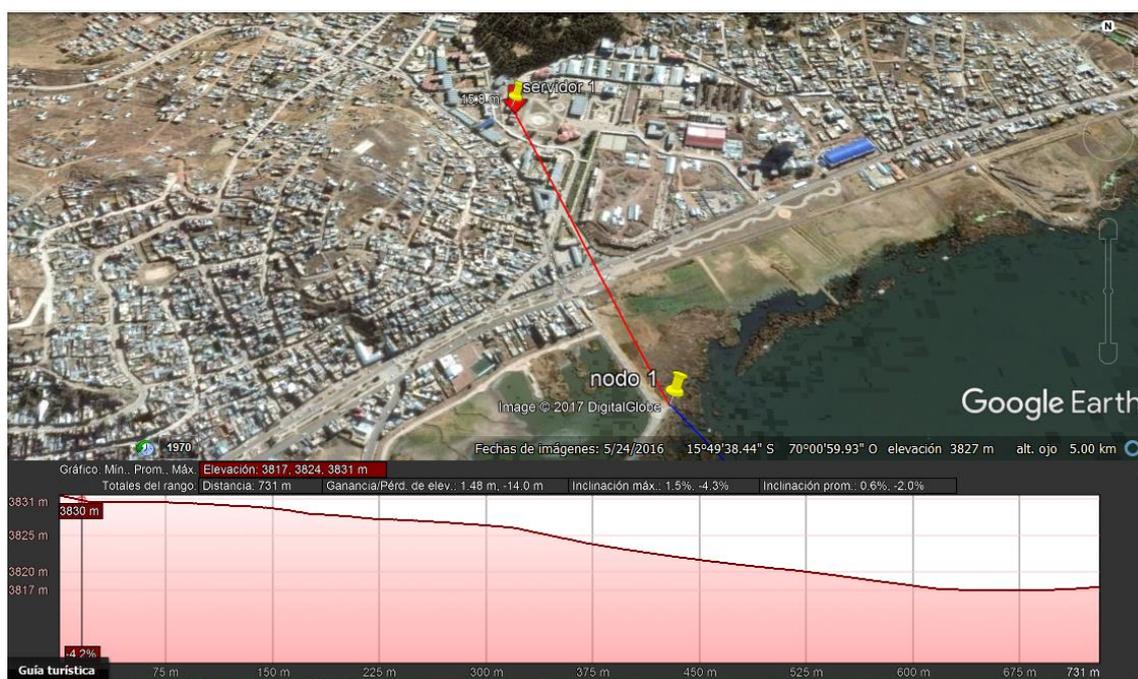
- ITU-T. (28 de Octubre de 2017). *Global Standards for the Internet of Things*.
Obtenido de Global Standards for the Internet of Things.:
<http://www.itu.int/en/ITU-T/techwatch/Pages/internetofthings.aspx>
- McEwen, A. &. (2014). *Designing the Internet of Things*. Reino Unido:
Chichester West Sussex.
- MONCHE, E. A. (2016). DEMOSTRADOR IOT-CLOUD EN TIEMPO REAL.
DEMOSTRADOR IOT-CLOUD EN TIEMPO REAL, 37.
- National Instruments. (12 de junio de 2017). *National Instruments*. Obtenido de
¿Qué es una Red Inalámbrica de Sensores?:
<http://www.ni.com/wsn/whatis/esa/>
- PELT, E. P. (2012). *MONITOREO DE LA CONTAMINACION DE LA BAHIA
INTERIOR DE PUNO*. Puno.

ANEXOS

**Anexo A: Lista de Comandos AT para la programación del módulo wifi
RS232**

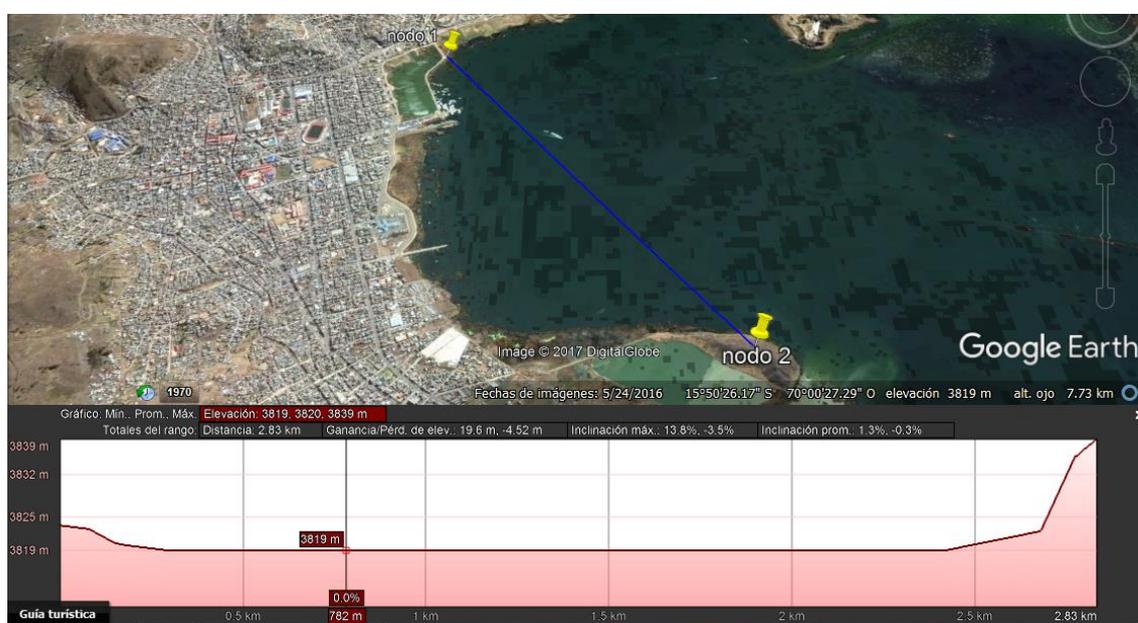
- AT+CWMODE_DEF=X. Configura modo de operación.
- AT+CWMODE?. Indica el modo de operación actual.
- AT+CWJAP_DEF="ssid","pass". Conecta a una red WiFi.
- AT+CIPSTA?. Lee la dirección IP, cliente y/o servidor.
- AT+CIPSTA=ip. Asigna una IP al modulo cuando se configura como cliente.
- AT+CWLAP . Proporciona una lista con las redes disponibles al módulo WiFi.
- AT+CWQAP . Desconecta el módulo de la red

Anexo B: Vista de enlace nodo 1 y servidor



Fuente: Google Earth Pro

Anexo B.1: Vista de enlace nodo 1 y nodo 2



Fuente: Google Earth Pro

Anexo C: CONFIGURACION DE SENSORES EN ARDUINO

```

#include <Adafruit_Sensor.h>
#include <DHT.h>
#define SSID "TITIKAKA" //Nombre de la Red Wifi a la que nos conectamos
#define PASS "titicaca2016" //Contraseña de la Red Wifi
#define IP "184.106.153.149" // thingspeak.com
//Sustituir [WRITE_API_KEY] por la obtenida en thingspeak.com sin []
String GET_TEMP = "GET /update?key=0TNGECPR7MEDKWC4&field1="; //Campo
de temperatura
String GET_HUM = "GET /update?key=0TNGECPR7MEDKWC4&field2="; //Campo de
humedad
#define DHTPIN 7 //Pin al cual se conecta el Sensor DHT11
#define DHTTYPE DHT11 //Tipo del Sensor
DHT dht(DHTPIN, DHTTYPE); //Declaracion del objeto DHT
void setup()
{

    pinMode(13,OUTPUT);

    dht.begin(); //Se inicializa sensor DHT
    Serial.begin(9600); //Se inicializa el puerto Serial para comunicarse
con el Modulo Wifi ESP8266
    Serial.setTimeout(5000); //Set el Timeout en 3 segundos para
Serial.find()
    Serial.println("AT\r"); //PROBAR CON \n\r

    delay(2000);

    if(Serial.find("OK")){ //Verificar si Modulo Wifi responde
        if(!connectWiFi()){ //Conectarse a la red
            Serial.println("NO SE PUDO CONECTAR A RED WIFI");
        }
    } else {
        Serial.println("MODULO WIFI NO RESPONDE");
    }
}

void loop(){
    float h = dht.readHumidity(); // Lee la humedad
    float t = dht.readTemperature(); //Lee la temperatura

    char bufferTemp[8]; //Crear buffer temporal para convertir float a
string
    String temperatura = dtostrf(t, 8, 3, bufferTemp); //Convertir de float
a String 8 = longitud de String, 3 = numero de decimales. (e.g. 1234.567)
    updateTemp(temperatura); //Actualizar temperatura
    digitalWrite(13,HIGH);
    delay(180000); //Thingspeak nos pide un MINIMO de 15 segundos entre
cada actualizacion
    //Si se hace mas pronto no se captura el Dato. Esperar 20 para estar
seguros
    ...//

```

```

    char bufferHum[8]; //Crear buffer temporal para convertir float a
    string
    String humedad = dtostrf(h, 8, 2, bufferHum); //Convertir de float a
    string 8 = longitud de String, 3 = numero de decimales. (e.g. 1234.567)
    updateHumedad(humedad); //Actualizar humedad
    digitalWrite(13,LOW);
    delay(180000); //Hacer una medicion cada 20 segundos
  }

void updateTemp(String temp){

    String cmd = "AT+CIPSTART=\"TCP\", \""; //Crear el comando para comenzar
    una conexion TCP
    cmd += IP; //IP del sitio a conectarse (thingspeak.com)
    cmd += "\",80"; //Puerto (80)
    Serial.println(cmd); //Crear conexion
    delay(2000); //Darle 2 segundos para responder

    if(Serial.find("ERROR")){
        return; //No se pudo conectar
    }

    temp.trim(); //Remover espacios e.g. "___30.00" >> "30.00"

    cmd = GET_TEMP; //Crear datos a enviar
    cmd += temp; //Agregar la temperatura leida
    cmd += "\r\n"; //Agregar caracteres de newline y Carriage return

    Serial.print("AT+CIPSEND="); //Indicar cuantos datos se enviaron por
    TCP
    Serial.println(cmd.length()); //Tamaño de los datos a enviar por TCP

    delay(2000);

    if(Serial.find(">")){
        Serial.print(cmd); //Enviar datos
    }else{
        Serial.println("AT+CIPCLOSE"); //No se pudo ejecutar, cerrar
        conexion TCP
    }
}

void updateHumedad(String humedad){
    String cmd = "AT+CIPSTART=\"TCP\", \"";
    cmd += IP;
    cmd += "\",80";
    Serial.println(cmd);
    delay(2000);

    if(Serial.find("ERROR")){
        return;
    }
}

```

...//

```
humedad.trim());

cmd = GET_HUM;
cmd += humedad;
cmd += "\r\n";
Serial.print("AT+CIPSEND=");
Serial.println(cmd.length());

delay(2000);

if(Serial.find(">")){
  Serial.print(cmd);
}else{
  Serial.println("AT+CIPCLOSE");
}
}

//Funcion para crear conexion Wifi. Regresa true si se conecta
exitosamente, si no false.
boolean connectWiFi(){

  Serial.println("AT+CWMODE=3"); //Configurar Modulo Wifi ESP8266 en
Modo 3 = Station + Access Point (Trasmitir y Recibir)
  delay(2000); //Darle tiempo para ejecutar comando

  String cmd="AT+CWJAP=\""; //
  cmd+=SSID;
  cmd+="\", \"";
  cmd+=PASS;
  cmd+="\"";

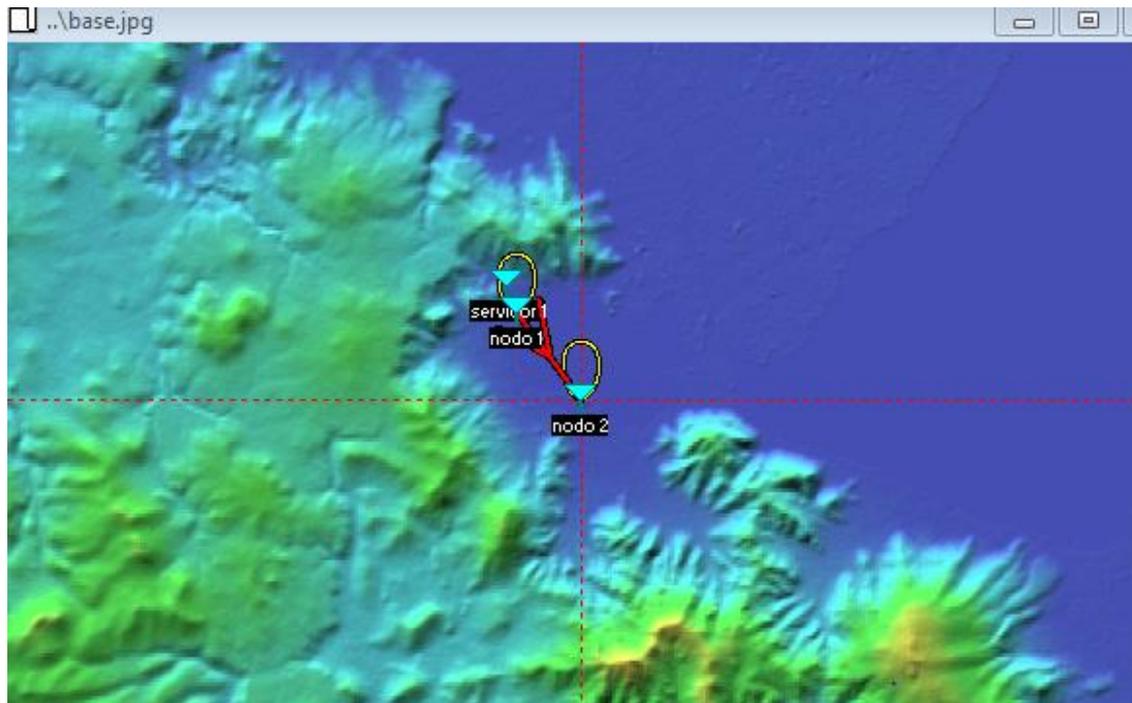
  //cmd = AT+CWJAP="SSID", "PASS"

  Serial.println(cmd);
  delay(10000); //Darle 10 segundos para conectarse

  if(Serial.find("OK")){
    return true; //Conexión existosa
  }else{
    return false; //No se pudo conectar
  }
}
```

Elaboración: Propia

Anexo D: diseño de enlace en Radio Mobile



Fuente: Elaboración Propia

Anexo E: Motas en el Lago Titicaca (malecón)



Fuente: Elaboración propia