



**FACULTAD DE INGENIERÍA, ARQUITECTURA Y
URBANISMO**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS
TESIS**

**PROTOCOLOS DE COMUNICACIÓN DE
INTERNET DE LAS COSAS PARA LA
TRANSFERENCIA DE DATOS APLICADOS A UN
SENSOR DE COLOR**

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO
DE SISTEMAS**

Autor:

Bach. Chucas Requejo Jose Fernando

ORCID: <https://orcid.org/0000-0003-0781-4034>

Asesor(a):

Dr. Tuesta Monteza Victor Alexci

ORCID: <https://orcid.org/0000-0002-5913-990X>

Línea de Investigación:

Infraestructura, Tecnología y Medio Ambiente

Pimentel – Perú 2022

APROBACIÓN DEL JURADO

PROTOCOLOS DE COMUNICACIÓN DE INTERNET DE LAS COSAS PARA LA TRANSFERENCIA DE DATOS APLICADOS A UN SENSOR DE COLOR

Bach., Chucas Requejo José Fernando
Autor

Dr., Tuesta Monteza Victor Alexci
Asesor

Mg., Atalaya Urrutia Carlos Willian
Presidente de Jurado

Mg., Celis Bravo Percy Javier
Secretario de Jurado

Dr., Tuesta Monteza Victor Alexci
Vocal de Jurado

Dedicatorias

Dedico mi tesis a mis Padres y Hermana
que siempre me han brindado siempre
el Respaldo y Amor para seguir adelante.

A mis familiares por sus consejos y enseñanzas
Para siempre seguir Adelante.

A Dios por Permitirme
Darme la Salud
Y el Entendimiento y poder
Cumplir Mis Metas.

Agradecimientos

A Dios por darme las fuerzas y la voluntad
de
seguir adelante a mi asesor de tesis
Víctor Tuesta Monteza
por sus enseñanzas y motivación en la investigación

Resumen

El internet de las cosas está enfocado en conectar millones de dispositivos y resolver problemas de pérdida de datos, evitar tiempo de inactividad en él envió de información y creando un mejor espacio brindando prioridad al utilizar información sensible de los usuarios, El propósito principal de investigación se basa en comparar 2 protocolos de capa aplicación de Internet de las Cosas (IOT), Demostrando la implementación de un proceso viable de envió de datos en donde el método propuesto inicia con 4 objetivos claves para evolución del proyecto, antes de ello es fundamental hacer una análisis de los antecedentes realizados por otros investigadores para evidenciar historial aceptable y seguro mediante fuentes fiables de investigación recomendadas por el asesor de la investigación, iniciando con la Selección de los Protocolos de Comunicación realizando un Análisis de otros Protocolos de comunicación de la misma capa aplicación IOT ,el segundo objetivo hace referencia a diseñar la arquitectura o estructura de red para efectuar la transmisión de data haciendo uso de las tecnologías, el tercer objetivo implementa los protocolos de comunicación utilizando la selección de los protocolos siendo probados individualmente para realizar la evaluación correspondientemente, el lenguaje de programación que se utilizó se desarrolló en el ambiente del IDE de Arduino y los medios de Host y los medios externos que se utilizaron también fueron documentados en la parte anexa del proyecto, el resultado más relevante al gestionar las pruebas el protocolo obtuvo de tiempo de envió de datos y en comparación con el otro protocolo arrojaron los siguientes resultados habiendo una diferencia y como último objetivo se mostraran los resultados convenientes para determinar la calidad de cada protocolo haciendo la comparación respectivamente siendo esta investigación un medio de como promover como fin único la investigación académica y ampliar el conocimiento de otros estudiantes a involucrarse por desarrollar nuevas tecnologías de información.

Palabras Clave:

Internet de las Cosas, MQTT, HTTP, ESP32, Protocolo Aplicacion, Broker, Topico

Abstract

The internet of things is focused on connecting millions of devices and solving data loss problems, avoiding downtime in sending information and creating a better space by giving priority when using sensitive user information. The main purpose of research is It is based on comparing 2 application layer protocols of the Internet of Things (IOT), demonstrating the implementation of a viable process of sending data where the proposed method begins with 4 key objectives for the evolution of the project, before that it is essential to make a background analysis carried out by other researchers to evidence acceptable and safe history through reliable research sources recommended by the research advisor, starting with the Selection of Communication Protocols by performing an Analysis of other communication Protocols of the same IOT application layer , the second objective refers to designing the architecture or network structure to carry out the transmission of data using the technologies, the third objective implements the communication protocols using the selection of the protocols being tested individually to carry out the corresponding evaluation, the programming language that was used was developed in the environment of the Arduino IDE and the Host media and the external media that were used were also documented in the annexed part of the project, the most relevant result when managing the tests the protocol obtained was data sending time and in comparison with the Another protocol yielded the following results, having a difference and as a last objective, the convenient results will be shown to determine the quality of each protocol, making the comparison respectively, this research being a means of promoting academic research as a sole purpose and expanding the knowledge of other students. to involve to develop new information technologies.

Keywords:

Internet of Things, MQTT, HTTP, ESP8266, Application Protocol, Broker, Topic

Índice

I. INTRODUCCIÓN	11
1.1. Realidad Problemática	11
1.2. Trabajos previos	12
1.3. Teorías relacionadas al tema	20
1.4. Formulación del Problema	39
1.5. Justificación e importancia del estudio	39
1.6. Hipótesis	39
1.7. Objetivos	40
1.7.1. Objetivo general	40
1.7.2. Objetivos específicos	40
II. MATERIAL Y MÉTODO	40
2.1. Tipo y Diseño de Investigación	40
2.2. Población y muestra	41
2.3. Variables, Operacionalización	41
2.4. Técnicas e instrumentos de recolección de datos, validez y confiabilidad	43
2.5. Procedimiento de análisis de datos	45
2.6. Criterios éticos	45
2.7. Criterios de Rigor Científico	45
III. RESULTADOS	46
3.1. Resultados en Tablas y Figuras	46
3.2. Discusión de resultados	49
3.3. Aporte práctico	49
IV. CONCLUSIONES Y RECOMENDACIONES	69
4.1. Conclusiones	69
4.2. Recomendaciones	69
REFERENCIAS.....	70
ANEXOS	73

Índice de Figuras

Figura 1. Billones de Conexiones de Internet de las Cosas.....	21
Figura 2. Capas del Modelo de Referencia OSI.	24
Figura 3. Ejemplo de Reenvío de Mensajes MQTT: Estándar frente a suscripción compartida.....	26
Figura 4. Secuencia de Comunicación de HTTP.	27
Figura 5. Dongle WiFi 802.11n	28
Figura 6. Arquitectura de Internet de las Cosas.....	29
Figura 7. Arquitectura de Sensor de Color TCS3200.....	32
Figura 8. Arquitectura de Sensor TC 3200.....	32
Figura 9. Combinación de Colores RGB.	33
Figura 10. Placa Arduino Mega 2560.	34
Figura 11. Imagen Raspberry PI 3 Model B+.....	38
Figura 12. Pruebas de Velocidad de Tr0ansferencia de Protocolo MQTT.....	47
Figura 13. Pruebas de Velocidad de Transferencia de Protocolo HTTP	47
Figura 14. Pruebas de Pérdida de Paquetes del Protocolo MQTT	48
Figura 15. Pruebas de Pérdida de Paquetes del Protocolo HTTP	48
Figura 16. Pruebas de Tiempo de Retardo del Protocolo MQTT.....	49
Figura 17. Pruebas de Tiempo de Retardo del Protocolo HTTP.....	49
Figura 18. Método Propuesto.	51
Figura 19. Diseño Físico de la Red.....	55
Figura 20. Descripción Pictográfico de la Arquitectura de la Red.	57
Figura 21. Estructura de Programación del Protocolo HTTP en Arduino.....	62
Figura 22. Estructura de Programación del Protocolo MQTT en Arduino	63
Figura 23. Se realiza la creacion del WebHost para la Pruebas y Envio de Datos	

Figura 24. Ejecución donde muestra la Lectura de los Colores mediante el Patrón de Salida de los Valores tomado por el Sensor de Color.....	65
Figura 25. Se realiza la Configuración del Programa Wireshark	65
Figura 26. Se selecciona el Dispositivo Wifi del cual se tomará los Datos del Trafico. 66	66
Figura 27. Se realiza la Toma y Configuración de la cantidad de paquetes, archivos,. .66	.66
Figura 28. Se realiza la Configuración para Obtener los Datos y Paquetes a Capturar.. 67	67
Figura 29. Al iniciar el Wireshark ya tenemos Trafico no solo del envío de Datos,.....	67
Figura 30. Captura de los Datos por medio del Aplicativo Wireshark	68

Índice de Tablas

Tabla 1. Recopilación de protocolos de Internet de las Cosas	25
Tabla 2. Muestra la escala de colores a partir de un rango de la longitud de onda 33	33
Tabla 3. Características Técnicas de los modelos Arduino	35
Tabla 4. Especificaciones Técnicas Arduino Mega 2560.....	35
Tabla 5. Definiciones de las Abreviaturas.....	36
Tabla 6. Especificaciones Técnicas Raspberry PI 3 Modelo B+	37
Tabla 7. Variables, Operacionalización.....	41
Tabla 8. Técnicas e instrumentos	43
Tabla 9. Criterios de Rigor Científico	46
Tabla 10. Criterios de la Evaluación	53
Tabla 11. <i>Evaluación y resultados de los protocolos de Internet de las Cosas Capa Aplicación (IOT)</i>	54

I. INTRODUCCIÓN

1.1. Realidad Problemática.

Las investigaciones que se resaltan en la actualidad el internet de las cosas propone nuevos protocolos de comunicación para poder evaluar distintos productos por medio de datos capturados mediante sensores de colores en conjunto con la tecnología Arduino se mostrará como poder enviar estos datos de manera automática hacia un servidor y estos datos recaudados puedan ser leídos y desde ese punto ejercer una opinión para determinar si esos datos son reales , estás tecnologías aún están en proceso de implementación.

Evidenciamos que existen protocolos de comunicación la cual es la arista principal para poder ayudar a enviar los datos para luego determinar si los datos capturados fueron enviados totalmente, como manera de aprovechar esta situación se puede cubrir un posible objetivo para clasificar el mango determinando el color, su estado actual como producto según la variedad de mango que tenemos en nuestro país. El sensor de color tiene problemas para poder calibrar los colores que deseamos mostrar para poder adaptarles protocolos de comunicación para los datos que se puedan enviar fuera de tal forma se hará una calibración a nivel de programación para determinar su mejor usabilidad .Al verificar que se han mostrado problemas en la capa aplicación se propone la evaluación de dos protocolos de la capa aplicación : MQTT y HTTP para evaluar y conocer cuál es el mejor de acuerdo a las Comparaciones y pruebas que se realicen.

Existen diversas enfermedades que afectan el desarrollo de los niños, una de ellas es denominada Trastorno del espectro autista que fue descubierto por (Jimenez, Rodriguez, & Tiparra, 1978). Esta enfermedad dificulta el aprendizaje, el desarrollo motor y las relaciones sociales, generando frustración en el niño. (Mejia, Ramirez, Jimenez, & Rosas, 2019).

Rojas, (2018), identificó cuatro efectos negativos en el aprendizaje, el primero es la concentración (SZNAJDLEDER, 2012).

Los textos anteriores son de ejemplo de cómo redactar y citar a un autor al inicio de párrafo, en medio del párrafo y al final del párrafo según APA. Se espera que la realidad problemática muestre el problema general, el problema

de ingeniería, las soluciones de ingeniería existentes, y se explique en ese contexto la contribución de esta exploración académica.

El propósito de la investigación es comparar dos protocolos de comunicación de la capa aplicación del internet de las cosas mediante un sensor de color utilizado para enviar datos a través de un dispositivo Arduino, cuyo objetivo es seleccionar los protocolos de la capa aplicación de Internet de las cosas para luego comparar cuál de los protocolos de comunicación de la capa aplicación rinde con más eficiente y quien cuenta con un mayor rango de rendimiento,

El proyecto de investigación se detalla en las siguientes líneas:

1.2. Trabajos previos.

Según **(Ramírez, 2017)** en su Resumen, Análisis de Rendimiento de protocolos de comunicación para plataformas de Internet de las cosas. Lleva a cabo una lista comparativa de los principales protocolos de comunicación que permiten la implementación de una plataforma IoT, para determinar su carga computacional, gastos generales y ancho de banda de red. Los problemas que soluciono fue determinar cuál es el mejor protocolo en la capa de aplicación para determinar con métricas el mejor de los cuatro protocolos, Los resultados se realizaron mediante pruebas de métricas: tiempo de trasmisión, envió, rendimiento de CPU, porcentaje de uso, bytes por transacción; mencionando que AMQT resalta en la capa de aplicación entre las 4 mejores resultantes que son MQTT, COAP Y XMPP. Describe un escenario evaluado en un servidor en donde se evaluó y arroja el resultado de tiempo de trasmisión de CoAP al enviar un archivo de 1MB demora aproximadamente dos minutos por tanto el protocolo MQTT y AMQP solo mantiene una demora aproximadamente de 6.4 segundos y 6.1 segundos respectivamente mientras que el payload se mantenga con un mínimo de 1 kilobyte CoAP Tarda aproximadamente en promedio a 238 milisegundos el cual es más cercanos a los tiempo dobles estimados tanto de

MQTT con 126 milisegundos y AMQP con 125 milisegundos, por consiguiente estos resultados se observó que los protocolos de estudio siendo los dos últimos evidenciaron un tiempo de transmisión con mucha similitud, cabe resaltar que después de una carga útil total de datos (payload) es decir de 512 kilobytes el protocolo AMQP sostiene una mayor velocidad en cuanto a MQTT .

relativamente el protocolos CoAP en la comparación hacia MQTT y AMQP es más lento sin embargo los paquetes que se envían demoran el doble y es por el motivo de que el cliente se demora en construir la data ,al enrutarse la data inicia desde el cliente conducido por el servidor hasta llegar al destino ,al respecto de generarse una carga masiva útil de datos (payload) en consecuencia a los 32 kilobytes ,el protocolo AMQP es el más estable y actúa de manera rápida a esto lo que resuelve es una mejora en cuanto a mayor rapidez y la diferencia que se busca va mostrándose a medida de que el archivo va creciendo (aproximadamente 1 megabyte) la diferencia con MQTT es aproximadamente de 1 segundo, con un payload se puede verificar que es menor de 32 kilobytes el tiempo consumido en el envío de paquetes de datos va en tendencia al alcance del protocolo MQTT.

Kevin Zhang, (2019), En su investigación, A Machine Learning Based Approach to Identify SQL Injection Vulnerabilities, realizada en Estados Unidos. Hace referencia que las páginas web están muy expuesta a recibir ataques de inyección SQL, por el motivo de recibir altas entradas de datos y eso hace que traiga consecuencias a largo plazo en base a clonación de código y errores de ejecución. Debido a esto fueron utilizados ocho prototipos de aprendizaje automático, como Decision Tree, Random Forest, SVM, Long Short Term Memory (LSTM), Logistic Regression, Multilayer Perceptron (MLP), Concurrent Neural Network (RNN) y Convolutional Neural Networks (CNN) para reconocer las principales debilidades que contiene al iniciar una inyección SQL en código Hypertext Preprocessor (PHP), donde se sustrajeron 18 validaciones tanto de entrada diferente y sanitización (IVS) evidenciando a dos números el primero es 1 si los caracteres son peligrosos y a 0 como No peligrosos.

Algunos métodos suelen definir sus propias validaciones y desinfecciones, para ello se creó una bolsa de palabras a base de las funciones dentro de un archivo, por la cual si no estaban relacionados a inyección SQL eran penalizados, de igual forma, para capturar el contexto de las características se utilizaron Word2vec, que se encargaba de encontrar relación entre palabras del archivo y las vulnerabilidades de inyección SQL. Los resultados obtenidos mostraron que la mejor precisión la obtuvo CNN con 95.4%, la mejor exhaustividad MLP con 63.7%, la mejor exactitud SVM con 95.4% y la mejor F-measure MLP con 0.746. Se logró detectar a nivel de indicadores las vulnerabilidades en inyecciones SQL, sin embargo, en trabajos futuros se pretende añadir gráficos de flujo de contexto (GFC) para mejorar el rendimiento de los modelos.

Sandy Suryo Prayogo et All,(2020) Realizo la Investigación, “El uso y rendimiento de MQTT y CoAP como protocolo de aplicación de Internet de las cosas utilizando NodeMCU ESP8266”, esta actividad se realizó en la Universidad de Jakarta en Indonesia comento el gran despliegue que ha tenido a lo largo de la última década hasta la actualidad el Internet de las cosas (IOT) y que su optimo rendimiento ha dado un gran resultado en base a crearse potenciales equipos de servicio optimizando muchos desafíos en la convivencia del uso de los protocolos y las mejoras al detectarse un mejor tiempo de envío de datos; se habla también de dos protocolos que son de MQTTy CoAP y estos dos admiten servicio de suscripción y publicación , en sus bases teóricas mencionan que sus arquitecturas son mediante un servidor por lo que es necesario elegir un protocolo adecuado para continuar con los procesos que se efectúan en el ensayo y pruebas por el cual proponen un esquema de trasmisión para enviar una cantidad de datos utilizando el dispositivo NodeMCU ESP8266 las pruebas hechas actúan en base al propósito de uso de cada protocolo experimentado y en la forma de comunicación a procesar los resultados esperados al respectos son que MQTT es más precios y confiable ya que cada medición depende del entorno de red en el que se realicen las pruebas.

Wolfgang Bziuk et All.(2020) En este artículo de investigación llamado “Sobre el rendimiento de HTTP en aplicaciones IoT: Un análisis de latencia y rendimiento”,

Realizado en la Universidad de Braunschweig en Alemania. Comenta que el nivel de desarrollo de IOT ha sido beneficiosa y de gran alcanza para crear objetos y dispositivos inteligentes con capacidades de procesamiento, detección y actuación a un nivel muy alto en base a ello y garantizar y que sean utilizados por todas las gamas de aplicaciones se estudiaron la variedad de protocolos de capa aplicación por el motivo de que n hay un protocolo estándar que represente a un único uso .Se realizo el estudio de una de ellas por sus características y definiciones HTTP al verse como un protocolo cliente y servidor se adecuaba a la infraestructura y compatibilidad de red existente y en base a el rendimiento que ejerce para la IOT , en esta investigación detalla el rendimiento y la latencia para HTTP/1.1 evaluando la canalización e impacto que provocaría mediante los factores de medición y en base a los resultados analizados se implementaran en el sistema de IOT con restricciones propuestas de latencia y análisis de datos.

Melvin Bender. et All. (2021) En este Articulo de investigación se propone “Evaluar el protocolo de comunicación de la capa aplicación MQTT evidenciando sus características de ser ligero de publicación y suscripción”, la investigación se realizó en la Universidad Técnica de Múnich en Alemania utilizado en el Internet de las Cosas (IOT). En este documento, evaluamos las implementaciones de MQTT de código abierto más populares, Mosquitto, HiveMQ, EMQX, VerneMQ, MQTT.js y Paho.

Las variables que se utilizaron y evaluación fueron interoperabilidad, consumo de recursos y latencia. se utilizó un diseño de prueba genérico independiente con la lengua de MQTT evidenciando los resultados de interoperabilidad, comunicación cliente servidor a los cuales el cliente solicitaba como requisitos básicos y se genere interoperabilidad el idioma de implementación es un factor crucial para el consumo de recursos y provoca diferencias considerables en la escalabilidad. Los resultados que arrojan permiten mencionar que la latencia implementada con las redes muestra dispersión y difieren al realizarse las pruebas, pero compensa en gran parte a la compensación entre el consumo de recursos y la latencia de red.

Ravi Kishore Kodali et All.(2017)En este artículo “Sistema de seguimiento meteorológico usando MQTT y SQLite”, se realizó la investigación en el Instituto Nacional de Tecnología Warangal en la India y se ha logrado contribuir al desarrollo de la comunicación y de acceder a la información sobre cierto lugar de forma remota y hacer de que las cosas alrededor nuestro se acciones de manera inteligente utilizando las tecnologías emergentes, se implementa un marco de referencia de Publicar datos mediante un sensor ESP8266 a un servidor de datos SQLite el cual está alojado a un Dispositivo Raspberry pi utilizando el Protocolo MQTT;el sensor esta ensamblado al microcontrolador NODEMCU ESP8266 para que se lean los valores. Se realizo una aplicación Python para detectar los valores del sensor y que estos sean publicados en Raspberry Pi y el Microcontrolador ESP8266 actúa como cliente MQTT y los datos del sensor se envían al corredor MQTT que se ejecutó en el módulo Raspberry pi. Asimismo, el servidor de datos SQLite también se alojó en el módulo Raspberry pi y los datos del ESP8266 se almacenan en formato JSON.los datos del servidor están formateados mediante un archivo HTML y se puede acceder a los datos formateados de forma remota. En el tramo final se realizó la implementación para acceder a la temperatura y humedad de cualquier lugar y de forma remota, para que los datos del sensor se puedan almacenar en Raspberry Pi la base de datos SQLite las particiona por separado y en diferentes Tablas en donde se ahorra mucha energía, memoria de almacenamiento y lo más importante es que el usuario recibe los datos desde cualquier parte del Mundo.

Yufei An. F. et al (2020), realizo la investigación” Borde de Inteligencia (EI): marco de detección de anomalías HTTP habilitado para Internet de las cosas (IoT)”. La investigación hace referencia al protocolo de aplicación utilizado en varias aplicaciones de IoT donde se estudió variedades de vulnerabilidades de seguridad. En la investigación se propuso un nuevo marco de detección de anomalías HTTP basado en inteligencia de borde para IoT. Conlleva a buscar nuevos métodos de agrupación y clasificación se utilizan para detectar de forma

rápida y precisa anomalías en el tráfico HTTP en base a otras diferencias e investigaciones existentes se basan en un servidor centralizado para realizar la detección de anomalías, con los avances recientes en inteligencia de borde, el marco propuesto distribuye todo el proceso de detección a diferentes nodos. Además, se propone un método de procesamiento de datos para dividir los campos de detección de datos HTTP, que puede eliminar datos redundantes y extraer características de los campos de un encabezado HTTP. La consecuencia de la ejecución de las pruebas demuestra que el plan propuesto conlleva a generar mejoras significativamente en cuanto a la velocidad y la precisión de la detección de anomalías HTTP, especialmente para problemas no conocidos.

Olumide Kayode & Ali Saman Tosun, (2019), En la investigación “Análisis de tráfico de IoT utilizando HTTP Proxy”, se desarrolló la pesquisa en la Universidad de Texas en Estados Unidos(UTEP); se Discutió sobre la era actual los datos y seguridad en los dispositivos es un comentario muy puesto sobre la mesa y el IOT y siendo tendencia y plato de discusión y preocupación principal de muchos usuarios e incluso fabricantes de estos mismo dispositivos emergentes ,debido a que esta misma data se vea expuesta y genere la desconfianza ,ya que es inobservada la gran cantidad de información que se procesa sin evaluar una tasa directa de tamaño ni tipo de información trasmitida en canales seguro e utilizados ,se exige una investigación más exploratoria, No todos los dispositivos IoT utilizan cifrado en su transmisión de datos y aquellos dispositivos que incorporan dicha medida de seguridad pueden verse comprometidos por la interceptación del tráfico generado a través del servidor proxy y su descifrado. Exploramos e investigamos los datos que se transmiten por seis dispositivos IoT representativos y analice los datos, utilizando un servidor proxy para capturar el tráfico HTTP y HTTPS.los resultados que se mostraron mediante dispositivos IoT transmite datos en plano texto mientras que otros utilizan cifrado, Información del usuario, la MAC dirección y la dirección IP se identificaron en nuestro análisis de datos. Se propuso que los dispositivos IoT no permitan conexiones proxy y implementar algoritmos de aprendizaje automático para detectar proxies usando información de conexión de red.

Abdur Rahman, Shanto Roy, M. Shamim Kaiser y Shahidul Islam, (2018), "Un marco S-MQTT ligero de varios niveles para asegurar la comunicación entre nodos de IoT de gama baja", se realizó esta investigación en la Universidad Verde de Bangladesh en Dhaka, comenta que Debido a la gran expansión y evolución ya mostrada en los últimos años en base a la tecnología a gran escala los dispositivos versátiles han dado lugar a la jerga Popular IOT, han evolucionado desde las (WSN) redes de sensores inalámbricos hasta ver casas Inteligentes utilizando la Domótica y la inteligencia artificial, en esta investigación se hace referencia y estudio a el Protocolo HTTP, se ha utilizado para el seguimiento y evaluación remoto de datos de gran volumen en base a elementos de detección, pero consume más energía, tiene una eficiencia de transmisión comparativamente menor y tampoco puede utilizar el ancho de banda del sistema de manera eficiente. Por lo tanto, los protocolos MQTT (Message Queuing Telemetry Transport), AMQP y CoAP son bastante competentes y de poder manejar la afluencia de dispositivos inalámbricos en un ancho de banda mínimo y condiciones de red limitadas. La protección igualmente es de importancia y de incertidumbre, ya que las aplicaciones de IoT reúnen datos con carácter privado y toleran el ingreso a muchos resultados de mando por medio de Internet.

En este análisis se discutió el análisis detallado de datos y problemas de seguridad de los dispositivos y presentar un modelo de seguridad mejorado con el fin de mejorar los problemas de seguridad. Se evaluó una versión segura del protocolo MQTT que modifica y mejora el protocolo MQTT existente basado en el cifrado basado en atributos de política de clave/texto cifrado (KP/CP-ABE) utilizando el criptosistema ligero Elíptica Curve. También introducimos un sistema de autenticación de varios niveles para una comunicación segura y una capa de seguridad complementaria y obstaculizar la sustracción de información sensible.

Hafsiya T.H Binet Rose y Kodakara, (2021), "Un dispositivo portátil de monitoreo de salud basado en la nube de IoT", en esta averiguación se realizó en el Departamento de ingeniería electrónica y de comunicaciones de Sahrdaya,

India. En donde explica la mayor dificultad a afrontar hoy en el universo es el virus corona más conocidos como COVID19. Muchas pruebas para Covid19 son costosas y están fuera del alcance de la persona promedio. Con la llegada de la pandemia a nivel mundial es de suma importancia la atención de la salud y la solución para dichas enfermedades infecciosas en IOT basado en la nube ,se propone realizar un sistema de vigilancia de salud por medios de los sistemas emergentes y descontrol de las ciencias médicas, el aumento de sensores utilizables y por lo tanto teléfonos inteligentes, El monitoreo remoto de la atención médica se ha desarrollado a tal ritmo que permitiría ayuda a prevenir la propagación de la enfermedad y hacer un diagnóstico preciso de la enfermedad del paciente; estado de salud incluso si el médico está lejos. En ese escenario se muestra un plan de control fisiológico portátiles cual será visualizado, que podrá comprobar constantemente el estado del paciente, frecuencia cardíaca, temperatura, nivel de oxígeno en sangre y sangre, presión, ya que los principales síntomas del coronavirus son fiebre alta, fatiga y dificultad para respirar. Los datos del sensor se transfieren directamente a la nube de IoT mediante Wi-Fi. Cualquier HTTP y MQTT protocolos utilizados en la nube de IoT para proporcionar a los usuarios y datos sensores cronometrados. Estos datos pueden ser fácilmente accedidos por casi todos los terminales inteligentes con navegador web. Los pacientes y su condición física también pueden ser monitoreada por la cámara. se sugiere un dispositivo de verificación y control continuo para verificar la condición del paciente y almacenar la información del paciente en el servidor en la nube, correspondencia remota basada en módulo Wi - Fi. Un sistema de monitoreo de salud remoto que usa IoT de forma remota que diagnostica pacientes usando la aplicación móvil/portátil, basada sobre estos valores obtenidos por personas autorizadas que han tenido acceso a los datos almacenados en cualquier plataforma IoT.

Padlan Alqinsi. et All, (2018),realizo la investigación "El Sistema de monitoreo de UPS basado en IoT que utiliza protocolos MQTT", el proyecto de investigación fue desarrollado en el Departamento de Ingeniería Eléctrica Bandung-Indonesia en donde se relata que el sistema de monitoreo de fuente de alimentación ininterrumpida (UPS) y hace mención al protocolo de

comunicación de capa transporte de telemetría de cola de mensajes (MQTT) haciendo uso de las habilidades y técnica de investigación se llega a la conclusión que permite solucionar los problemas de monitoreo de UPS evidenciando su potencial infraestructura., en esta investigación se analiza la prueba del tamaño pequeño de los datos de los mensajes mediante el uso del protocolo MQTT en las comunicaciones basadas en IoT, el sistema descrito en este documento utilizó un microcontrolador Arduino conectado a Internet a través de un escudo Ethernet. Este sistema estaba destinado a mostrar datos de monitoreo de UPS en tiempo real en una página web almacenada en raspberry pi que funciona como servidor web, corredor MQTT, suscriptor MQTT y base de datos. El parámetro UPS se puede monitorear usando una aplicación basada en la web. Hubo algunas diferencias en los datos obtenidos del sensor con los resultados de medición de los instrumentos de medición especificados para cada parámetro: la diferencia de voltaje de entrada fue igual a 0,20 %, mientras que las diferencias de voltaje de salida, potencia de salida y corriente de salida alcanzaron 1,34 %, 0,17 % y 20%, respectivamente.

1.3. Teorías relacionadas al tema.

1.3.1. Internet de las Cosas

Según (Ashton, 1999), es un tecnólogo visionario experto que se inició en la transformación digital, CEO de Laboratorio de Investigación Auto-ID, fue fundador y emblema de la palabra Internet de las cosas (IoT), en el Instituto de Tecnología de Massachusetts (MIT), es en donde se gestionaban las investigaciones en el campo de la Identificación de servicios de radiofrecuencia (RFID) y tecnologías con sensores. El gobierno chino y el consejo Nacional de Inteligencia (CNI) hicieron la referencia que es una importante tecnología evocada a ser desarrollada en el siglo XXI.

Cada objeto puede identificarse, tiene la facultad de poder hacer la detección, procesamiento y comunicación. La naturaleza que esta está al alcance y se puede tener en cualquier lugar, los servicios en IoT recoge la data y luego puede compartirlos para uso público y privado es muy importante también garantizar que los datos sean seguros en cuanto a

integridad y sean tramitados con absoluta confidencialidad, así como la autenticación de cada dato y de la privacidad en IoT. Según (Margery, 2010) menciona a la interconexión digital de cosas u objetos en Internet. Según Gartner, en el año 2020 el mundo tendrá aproximadamente 26 mil-millones de conexiones a Internet de las cosas (Gartner, 2010).

Se evidencia en la siguiente tabla de acuerdo con el estudio realizado por (Gartner, 2010) en donde se muestra que para el 2022 se aproxima billones de conexiones.

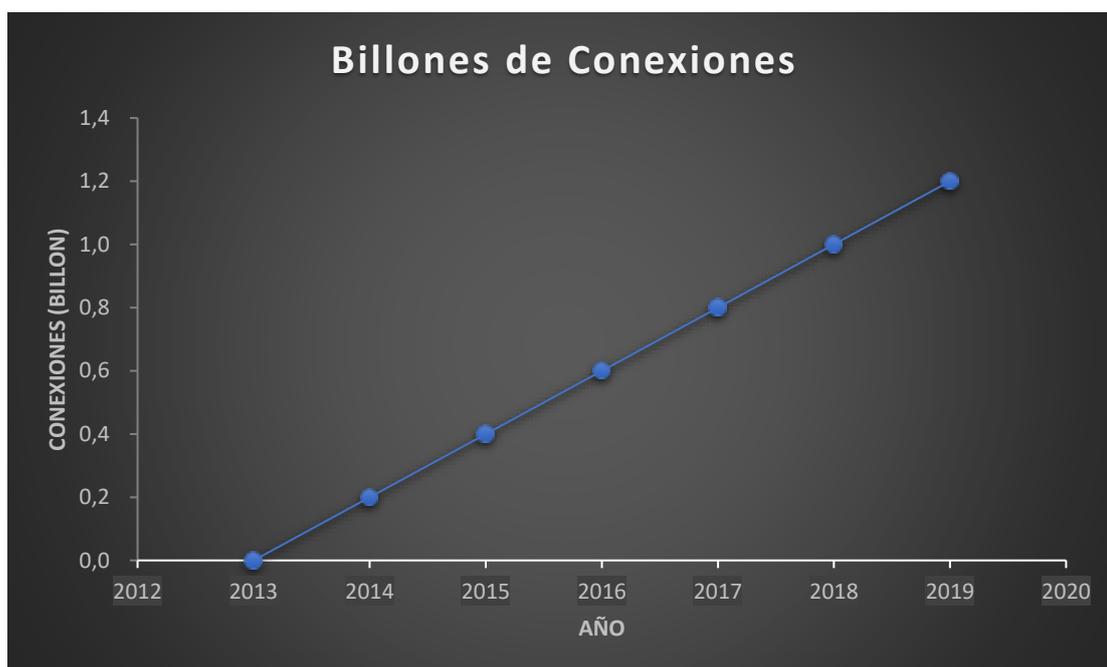


Figura 1. Billones de Conexiones de Internet de las Cosas.

Fuente:(Gardner, 2010)

1.3.2. Identificación de Radio Frecuencia (RFID)

Según el autor (Ashton y Sarma, 1999) mencionan que es una tecnología capaz de almacena/captura datos electrónicamente, Es circuito integrado y se puede utilizar como una antena. Se tiene diferentes tipos de etiquetas de RFID y son muy conocidas como tags.

activas: Incluye un sistema de permite suministrarle energía (batería, un panel solar) opera en transmitir una señal de geolocalización. Mantiene rangos fidedignos aproximadamente de 10 metros y opera en un ámbito (agua, metal).

pasivas: hacen uso de poder tomar energía de campos electromagnéticos generado por el lector para compartir data que se almacena en memoria es decir es un enrutador de datos. Posee un alcance entre 10mm a 6m. el tag tiene la capacidad de soportar condiciones de ambiente extremo.

semi-pasivas: Se caracteriza por tener una batería que alimenta el circuito hasta que se mantenga una comunicación con el lector, hace uso de un campo eléctrico para distribuir energía para enviar data.

Solo lectura: Es un tag solo permite la acción de Lectura.

Múltiple lectura escritura: El tag solo permite al Dispositivo permite solo escritura/lectura de data, no se puede generar otro tipo de actividad.

1.3.3. Comunicación de Campo Cercano - NIC

Tecnologías capaces de conectar dispositivos electrónicos por dos conductos de manera simultánea y segura y es de interés para dispositivos emergentes y en cuanto a los consumidores generan transacciones de pago y operan teniendo acceso a contenido virtual de tal manera es parte del sistema de comunicación (Pérez, 2017).

Utiliza el Estándar ISO 18000-3 con una Frecuencia no menor aproximada De 13.56 MegaHerz Abarcando aproximadamente 10 centímetros siendo Una velocidad de Transferencia de 100 kbps aproximadamente hasta 420 kbps.

1.3.4. Modelo OSI TCP/IP

La Organización Internacional de Estándares ISO, gestiona el uso y creación de un modelo para gestionar una arquitectura de red orientada a niveles: el modelo de Interconexión del sistema abierto (OSI), un estándar de comunicación de sistemas abiertos. Estructuración de los servicios de red y son siete capas, la primera está más cercana al medio físico y la séptima capa está más cercana a las aplicaciones que gestionan usuario, Cuando un agente necesita transferir data a una unidad de destino, el sistema de red genera información de control para cada uno de los servicios que maneja la utilización de la red para enrutar la orden.

El modelo OSI se define con las siguientes capas:

A sujeción daremos la definición resumida de las capas del modelo u norma OSI.

a. Físico: Observar permanentemente la línea de comunicaciones, define como y que medio se va a utilizar (puertos, cables, etc.).

Intervienen en ellos: Transmite bits por canal, Velocidad y Conductores Full Dúplex

b. Enlace: Proteger la información y verificar la integridad de esta; es decir, que la información transmitida llegue sin errores.

La capa Toma el control físico y lo pasa libre de errores se puede saber cuándo inicia y termina la trama y permite retransmitir si hay daño teniendo cuidado con la generación de duplicados.

c. Red: Establece comunicación con el host, coloca encabezados para el enrutamiento (Direccionamiento IP) la cual interviene el Control de subred, Enrutamiento y Broadcast.

d. Transporte: Protocolo TPC/IP; divide la información en franjas secuenciales (Tramas), Requiere enumerar las tramas con el fin de que el receptor pueda ordenarlas.

Recibe del nivel de sesión, pasa a pequeñas unidades y coloca en el nivel de red conexión libre de errores, punto a punto entregando en el orden en que fueron enviados.

e. Sesión: ID de la comunicación controla y numera la sesión (Login) controla el "Quién está haciendo "que" también permite a varios equipos establecer sesiones entre ellos Login remoto, Transferencia de archivos y sincronización.

f. Presentación: Código que se va a utilizar y encriptar si se desea Sintaxis y semántica de la información transmitida, Codificación de los datos (ASCII, EBDIC).

g. Aplicación: Programas que corren en el host a nivel del usuario
Interviene Muchos protocolos, el Mail y Transferencia de archivos.

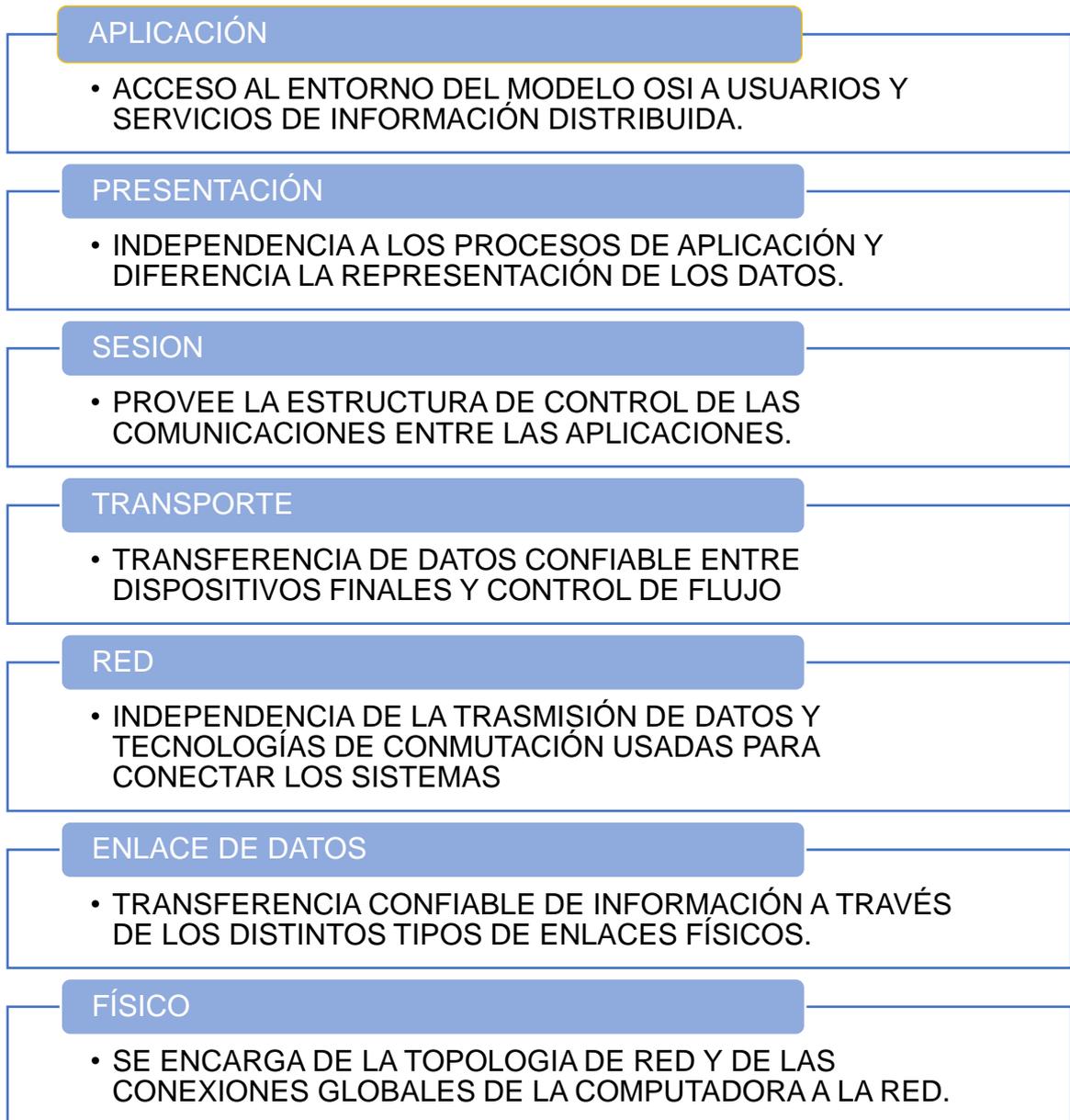


Figura 2. Capas del Modelo de Referencia OSI.

Fuente:(Stallings, 2009)

Tabla 1. Recopilación de protocolos de Internet de las Cosas

PROTOCOLOS CAPA APLICACIÓN		DDS	CoAP	AMQP	MQTT	MQTT-SN	XMPP	STOMP	HTTP
SERVICIO DESCUBRIMIENTO		Mdns			DNS-SD				
INFRAESTRUCTURA PROTOCOLOS	PROTOCOLO DE ENRUTAMIENTO	RPL							
	CAPA RED	6LOWPAN			IPv4/IPv6				
	CAPA ENLACE	IEEE 802.15.4							
	CAPA FISICA	LTE-A	EPCglobal	IEEE 802.15.4	Z.WAVE				
PROTOCOLO INFLUENCIAL		IEEE.1888.3,IPSec						IEEE 1905.1	

Fuente: Elaboración Propia basada en Fuhaga & Aledhari,2015;

1.3.5. Protocolos de Comunicación

Los protocolos de comunicación son importantes para proyectos que sostienen comunicación constante con servidores o dispositivos inteligentes, es decir se desarrollaron protocolos con la intención específicamente de facilitar la transmisión de data más rápida y fidedigna.

1.3.6. Protocolos de aplicación

1.3.6.1. Protocolo MQTT

Este protocolo significa y a la vez represente un mensaje de transporte de telemetria en cola ,utiliza un publicador/suscriptor diseñado para comunicación ligera de maquina a máquina M2M ,originalmente fue desarrollado por IBM y ahora es un estándar, la arquitectura MQTT tiene un modelo Cliente/Servidor donde cada sensor es un cliente y se conecta a un servidor, el servidor se conoce como un corredor(BROKER) mediante comunicación Tcp.cada mensaje enviado es Publicado por un Topic que es

una cadena de texto y los clientes pueden suscribirse a varios Topics (Nikolov,2020).

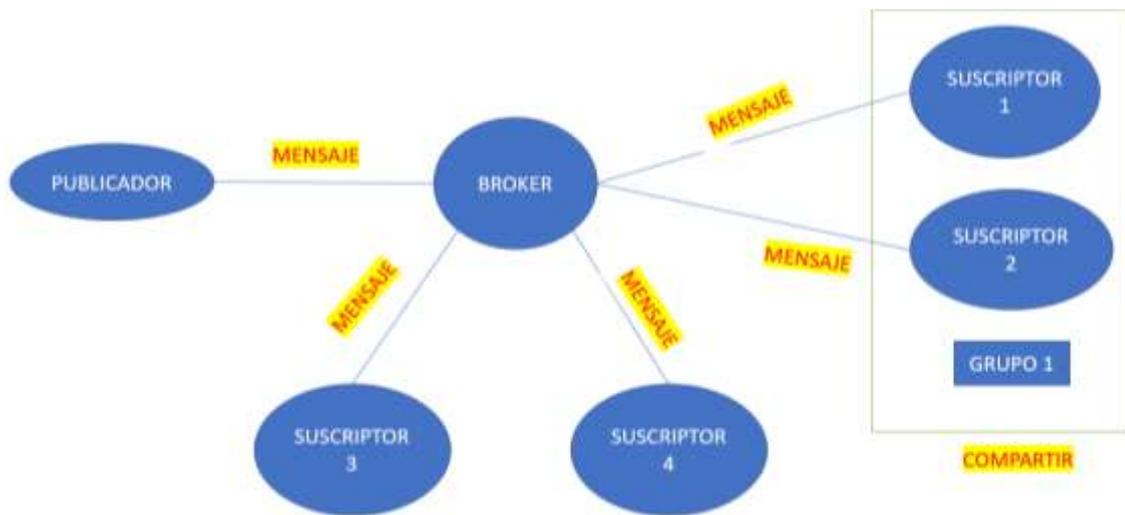


Figura 3. Ejemplo de Reenvío de Mensajes MQTT: Estándar frente a suscripción compartida.

Fuente:(Milica Matić, Marija Antić y Oros 24-25, 2020)

1.3.6.2. Protocolo HTTP

HTTP conocido como Protocolo de Transferencia de Hipertexto nos permite accionar una petición de datos o recursos que puedan ser documentados en HTML, es la base de cualquier intercambio de datos sobre la web y mantiene una estructura cliente/servidor, En otras palabras, es más seguro construir un dispositivo IoT que pueda iniciar conexiones, no recibir, se define con operaciones como GET, POST, PUT, DELETE y otros métodos. Desglosa las solicitudes y respuestas de datos de gran tamaño en fragmentos administrados por la red, está diseñado para la comunicación entre 2 sistemas solo a la vez. En la mayoría de las aplicaciones de IoT, hay una gran cantidad de sensores que generan datos al mismo tiempo y quieren enviarlos al servidor al mismo tiempo y en la práctica no es factible la comunicación de uno a muchos en el contexto de un numero de sensores y el servidor.

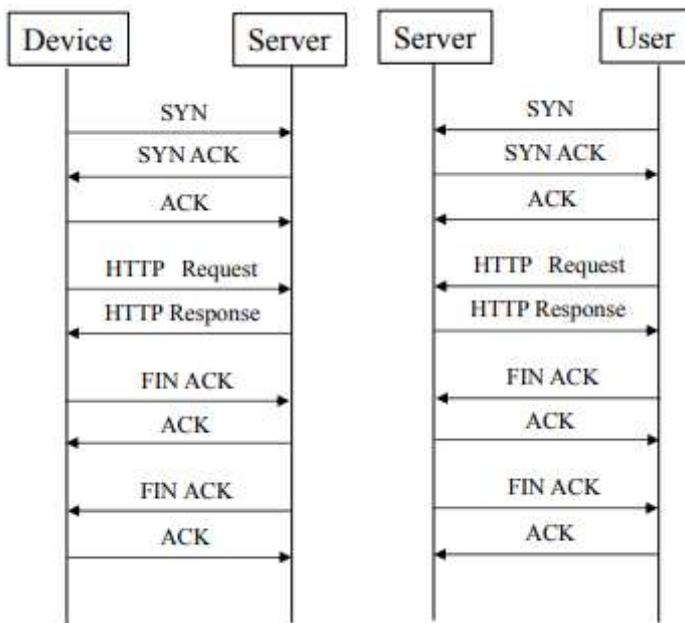


Figura 4. Secuencia de Comunicación de HTTP.

Fuente: (Tetsuya Yokotani ,Yuya Sasaki,1996).

1.3.7. Protocolos de comunicación inalámbricos

Una de las apariencias generales que más impresión tiene en brindar las redes o dispositivos emergentes empleados para los protocolos en sus capas, Se conoce la realidad de protocolos estandarizados (IEEE 802.15.4), pero en cuanto crece el mercado se siguen creando nuevas propuestas de crear más protocolos y en demanda de crear más aplicaciones lo necesitan y teniendo en cuenta las fuentes de energía muy limitadas. A continuación, mostraremos los siguientes estándares de comunicación inalámbrica:

1.3.7.1 Wifi: Estándar de comunicación IEEE 802.11 Es una extensión de comunicación entre computadores a nivel mundial, tecnología que abarca un inmenso ancho de banda a nivel mundial.

1.3.7.2 BlueTooth: Estándar Bluetooth capaz de transmitir voz y datos con distintos materiales mediante un nexo de radiofrecuencia. Su requerimiento es mínimo que el WiFi y su Modelo es usado en telefonía móvil.

- 1.3.7.3 802.15.4/Zigbee:** Trazado para redes inalámbricas de sensores, está orientado a dispositivos de bajo costo y ayuda en un reducido consumo de energía y en un menor uso de hardware y software.
- 1.3.7.4 6LowPAN:** estándar capaz de realizar una conexión con dispositivos en un área de trabajo a nivel de red inalámbrica tenga comunicación con otras direcciones ip.
- 1.3.7.5 Tecnologías inalámbricas sub-GHz:** El único objetivo de esta tecnología es el bajo consumo de transmisión como en la recepción de datos y su mayor privilegio es que tiene un alto radio de alcance y soportan incidencias climatológicas o meteorológicas.



Figura 5. Dongle WiFi 802.11n

Fuente:(IEEE, 2001)

- 1.3.7.6. Network Time Protocol (NTP):** Protocolo de red que permite coincidir relojes en sistemas de conmutación conducido por paquetes en redes de datos.NTP es uno de los protocolos de Internet más viejos en uso. Un equipo de voluntarios todavía lo desarrolla y mantiene. Usa una versión modificada del algoritmo de Marzullo para seleccionar con exactitud los tiempos del servidor y está designado para mitigar los efectos de la latencia variable de la red. Puede conseguir una precisión de decenas de milisegundos a través de Internet público y de milisegundos en redes LAN.
- 1.3.7.7. Secure Shell (SSH):** Protocolo de capa red con funciones criptográficas gestiona comunicación segura de archivos, inicio sesión seguro desde línea de comandos, ejecución remota de comandos y otros agentes seguros en la red de 2

computadores unidas a la red, se une por un túnel seguro sobre una red vulnerable a un cliente/server utilizando SSH. SSH SU conexión es mediante un BeagleBone a través de una red Ethernet o WiFi para su posterior distribución.

1.3.7.8. Secure CoPy (SCP): es un mecanismo donde transferimos un conjunto organizado de unidades de información (ficheros) entre 2 dispositivos puede ser uno local y el otro remoto., pero cada una tiene una función específica y ya está certificadas cada capa mencionada tiene una función diferente en donde define sus protocolos. Se muestra con más detalle en la figura 6 la arquitectura.

1.3.8 La Arquitectura de tres capas

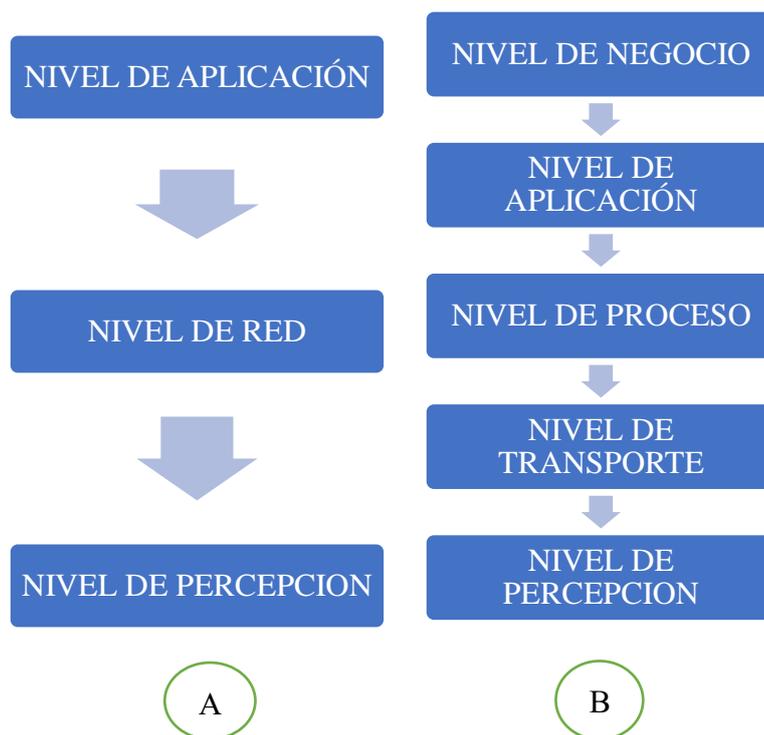


Figura 6. Arquitectura de Internet de las Cosas.

Fuente:(Granjal, and Silva, 2013)

La Figura N° 6 corresponde a la arquitectura de internet de las cosas. La parte del lado izquierda corresponde a 3 capas la capa Percepción hace referencia al

nivel físico por ejemplo cuando se capturan datos por medios inalámbricos desde los sensores; el segundo nivel refiere al tercer nivel donde el usuario genera transacciones.

- 1.3.8.1. La capa percepción: Refiere a los sensores partícipes de capturar data.
- 1.3.8.2. La capa red se encarga de conectar el dispositivo a otras cosas inteligentes, Dispone de las herramientas necesarias para transmitir data entre dispositivos (o servidores y dispositivos de red),
- 1.3.8.3. La capa de aplicación aquí se gestiona las aplicaciones de usuarios por ejemplo aplicaciones agua, gas o electricidad hasta las aplicaciones logísticas para empresas que se basan en poder optimizar recursos y ahorrar tiempo de gestión.

La Arquitectura describe la lógica de la tecnología es decir los datos recaudados de los dispositivos tangente trasmiten por otros medios de red que procesan data en aplicaciones que son de uso doméstico, pero en la vida real a la hora que se desarrollan aplicaciones reales se necesitan un mayor nivel de actuación a ocupar más capas u otros conceptos para definir

1.3.8.4. La arquitectura de cinco capas

Esta arquitectura tiene 2 capas, la capa percepción y aplicación, los tres restantes se muestran aquí:

La capa de Transporte se hace cargo de resolver la comunicación a nivel de red utilizando por ejemplo las redes 3g/4g, Wifi, Bluethoo Cuando ya se obtiene unos resultados de datos entre dispositivos por ejemplo en la capa transporte de red de comunicación utilizada se realiza el proceso de abstracción y se captura los datos. En esta capa

se ubican los servicios de procesamiento de data por ejemplo bases de datos, computación en la nube. La última capa es de Negocio aquí se resuelven problemas en cuanto a abstracción de data como otros modelos de negocios la privacidad y autenticidad de datos de usuarios es importante en cuanto a gestión de aplicaciones.

1.3.9. Sensores de color

Se utilizó el sensor como transductor, que incluía una serie de fotodetectores para actuar como sensor.

El módulo tiene cuatro entradas digitales para escalar la frecuencia de salida del sensor y seleccionar un filtro RGB. Este transductor combinar fotodiodos de silicio con un convertidor de frecuencia a corriente en un circuito integrado y la salida del sensor es una onda cuadrada con un ciclo de trabajo del 50%, cuya frecuencia es proporcional a la intensidad de la luz. Es un convertidor de luz a frecuencia que lee una matriz de 8x8 fotodiodos, de tal manera que 16 fotodiodos tienen filtro azul, 16 fotodiodos tienen filtro verde, 16 fotodiodos tienen filtro rojo y 16 fotodiodos son sin filtro.

Las Especificaciones técnicas del sensor de color se describen a continuación:

mantiene un alto porcentaje de conversión de luz a frecuencia de salida en color y tiene comunicación directa con un microcontrolador y su voltaje de funcionamiento es de 2.7 Voltios a máximo 5.5 Voltios.

Tiene un rango del porcentaje de error aproximado del 0,2% a 50 kHz.

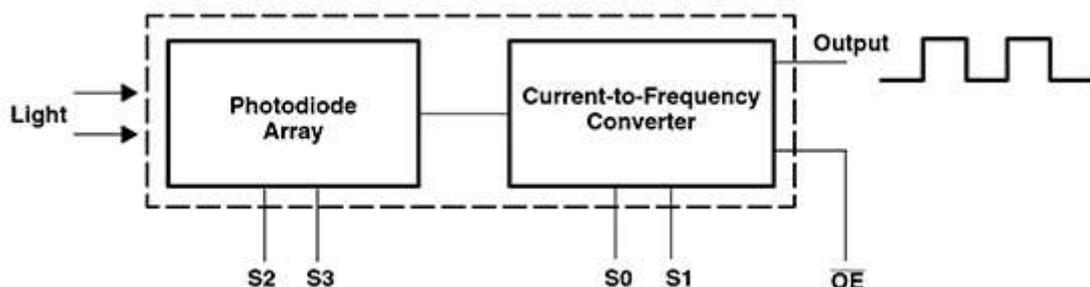


Figura 7. Arquitectura de Sensor de Color TCS3200.

Fuente: (Pérez, 2017)

Frecuentemente el sensor de color con modelo TCS-3200 se distribuye en módulos que incorporan cuatro leds de luz blanco y el principal objetivo de ambas medidas es minimizar los efectos de la iluminación ambiente en la medición, el sensor se tiene que calibrar para poder captar los colores RGB de un objeto, se puede usar para medir los colores básicos por ejemplo reconocer el color de una tarjeta u objeto y guiar a un robot en un tramo de recorrido. El sensor de color tiene cuatro entradas digitales que consta de 4 pines digitales (S0, S1, S3 y S4), Para conectarlo a Arduino necesitaremos emplear al menos 3 pines digitales de las cuales el primer módulo conecta los pines Gnd y Vcc del sensor.

Los dos primeros pines, la entrada So y la entrada S1 permiten controlar la frecuencia de salida y la desactivación del módulo. Los conectamos a dos salidas digitales y el soporte máximo de voltaje que puede recibir es de 5 Voltios. La entrada del pin S2 y el pin S3 seleccionan el color al cual se medir (Arduino, 2010).

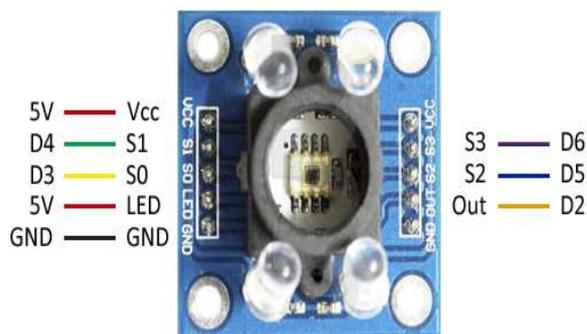


Figura 8. Arquitectura de Sensor TC 3200.

Fuente:(Arduino, 2010).

1.3.10. Espacios de color RGB

Este espacio que es básico en la evolución de los colores se encuentra una escala de luces que se visualizan por medio de los colores primarios como el Rojo, Verde, Azul, Estos colores son fundamentales porque de estos se dispondrá de más tonos de colores, Es decir que al mezclarlos

producirá un color diferente y el resultante vendría a convertirse en un círculo cromático.

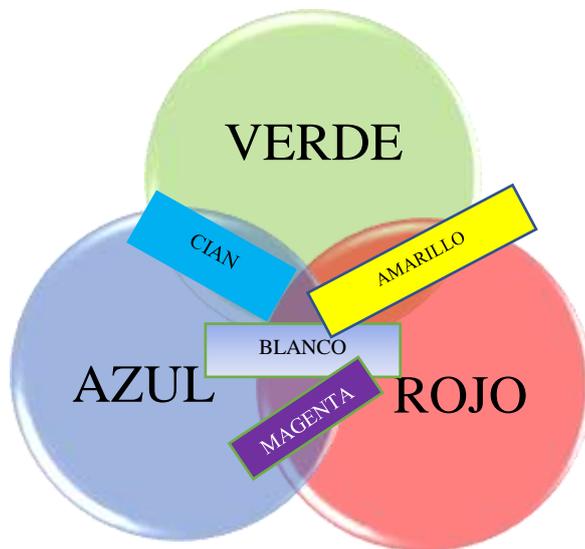


Figura 9. Combinación de Colores RGB.

Fuente:(Vega, 2006)

Tabla 2. Muestra la escala de colores a partir de un rango de la longitud de onda

COLOR	RANGO DE LONGITUD DE ONDA(NM)
ROJO	625-740
NARANJA	590-625
AMARILLO	565-590
VERDE	520-565
CIAN	500-520
AZUL	435-500
VIOLETA	380-435

Fuente:(Vega, 2006)

1.3.11. Arduino

Es uno de los hardware más usados a nivel mundial para configuración de objetos electrónicos, Arduino inicio con una plataforma electrónica, sigue siendo una placa, pero anteriormente empieza sosteniendo 2 puertos con uno de entrada y otro de salida, aplica la programación más básica del mundo basada a código abierto orientado a la a funciones procesamiento

y escritura. Es decir, se puede visualizar por medio de una pantalla la programación que se está ejecutando en tiempo real en el IDE.

Las Placas Arduino se usan como elementos electrónicos, electrodomésticos para controlar los encendidos de motores de arranque y también para convertir los datos analógicos a digitales y servir para pequeños proyectos o caseros de investigación portátil.

El Lenguaje de programación con el que trabaja es compatible con muchos lenguajes de programación como Visual Basic para programar soluciones sistemáticas dentro del Ambiente de Windows.

Puedes usar Arduino para crear objetos interactivos, leyendo datos de una gran variedad de sensores y controlar multitud de tipo de luces, motores y otros actuadores físicos, Estos Proyectos pueden ser Autónomos o comunicarse con un programa (software) que se ejecuta en un ordenador, Puedes montarla tú mismo o comprarla ya lista para utilizarla el software se puede descargar desde la página (Arduino, 2010).



Figura 10. Placa Arduino Mega 2560.

Fuente:(Arduino, 2010).

Tabla 3. Características Técnicas de los modelos Arduino

Modelo	Arduino UNO	Arduino Mega2560	Arduino Leonardo	Arduino Due
Microcontrolador	Atmega 328 8bits	Atmega 2560 8bits	Atmega 32u4 8bits	ARM SAM3X86 32bits
Memoria RAM	2KB	8 KB	2.5 KB	96kb
Memoria eeprom	1KB	4 KB	1 KB	0kb
Memoria Flash	32KB	256 KB	2 KB	512kb
Pines Analógicos	14	54	28	54
Pines Digitales	6	16	12	12
Pines con Interrupciones	2	6	2	-
Pines PWM	6	15	7	12

(Arduino,2010)

Tabla 4. Especificaciones Técnicas Arduino Mega 2560

Especificaciones Técnicas	
Microcontrolador	Atmega2560
Pines Digitales	54
Pines Analógicos	16
Corriente DC de cada Pin	40 mA
Corriente DC en 3.3v	50 mA
Memoria Flash	256 KB
SRAM	8 KB
EEPROM	4 KB
Frecuencia de Reloj	16 MHz
Voltaje de Alimentación	5 V

Fuente: (Arduino, 2010).

La alimentación del Arduino se puede dar desde un transformador o una batería externa, por tanto, los voltajes tanto de entrada van desde el rango de 6 Voltios

hasta aproximadamente 20 Voltios, por tanto, el mínimo de 7 Voltios sería factible que no sea suficiente con los 5V pueda mantener el requerimiento del voltaje y tenga un máximo rendimiento. Es decirlo más recomendable sería mantener un voltaje requerido de entre 7 voltios hasta aproximadamente 12 Voltios. Para más detalle se muestra en la tabla N°5 las abreviaturas de los pines:

Tabla 5. Definiciones de las Abreviaturas

Nombre de pin	Descripción
VIN	suministra alimentación desde fuente externa
5V	Se pueden obtener 5 Voltios pasando por regulador
3V3	Se puede obtener 3.3Voltios con la corriente máxima de 50 mA
GND	Toma de masa del circuito o tensión de referencia de 0 Voltios
IOREF	Proporciona la referencia de tensión con la que opera el microcontrolador para cualquier shield o elemento acoplable

Fuente:(Arduino, 2010).

1.3.12. RabbitMQ

Fue creada en abril del 2010 por la división SpringSource de VMWare El código fuente está liberado bajo la licencia Publica de Mozilla Firefox, es un software de negociación de mensajes de código abierto que implementa el estándar Protocolo avanzado de cola de mensajes (AMQP) se desarrolló en Erlang y utiliza un framework de Plataforma Abierta Telecom (OTP) de la cual con este mismo adquiere la capacidad de distribución

El proyecto RabbitMQ consta de diferentes partes: Tiene Pasarelas para los protocolos HTTP, XMPP y STOMP.MQTT; Utiliza Bibliotecas de clientes para java y el framework .NET y utiliza un plugin “pala “que es importante decir replica mensajes de un lado a otro.

1.3.13. Raspberry

Según (Oracle, 2009) un Raspberry es un ordenador de bajo coste y tamaño reducido ya que es un dispositivo muy ligero y fácil de llevar,

también permite conectarse a un televisor y teclado para interactuar exactamente igual a cualquier pc de escritorio u laptop. Se Originó en el Reino Unido en un Proyecto ambicioso de Fundación Raspberry y Oracle Academy en el Año 2009, con la intención de lanzar una estación Meteorológica para escuelas en un experimento Global, en cada escuela recibieron un paquete conteniendo el hardware Raspberry y para que los estudiantes den uso creando una estación Meteorológica con el único objetivo de obtener conocimientos computacionales, meteorología, geografía. Fueron más de mil kits se dieron de manera gratuita y académica entre el rango de edad de 16 y 11 años, de una manera hicieron contribuir a esta causa para motivar a los estudiantes y así ganar experiencia (Arduino, 2010).

Tabla 6. Especificaciones Técnicas Raspberry Pi 3 Modelo B+

PROCESADOR	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC
FRECUENCIA DE RELOJ	1,4 GHz
MEMORIA	1GB LPDDR2 SDRAM
CONECTIVIDAD INALÁMBRICA	2.4GHz / 5GHz IEEE 802.11.b/g/n/ac Bluetooth 4.2, BLE
CONECTIVIDAD DE RED	Gigabit Ethernet over USB 2.0 (300 Mbps de máximo teórico)
PUERTOS	GPIO 40 pines HDMI 4 x USB 2.0 CSI (cámara Raspberry Pi) DSI (pantalla táctil) Toma auriculares / vídeo compuesto Micro SD Micro USB (alimentación) Power-over-Ethernet (PoE)



Figura 11. Imagen Raspberry PI 3 Model B+.

Fuente:(Arduino, 2010)

1.3.14. NodeMCU ESP8266

Es una placa que posee su propio núcleo y permite una conexión dentro de una plataforma de desarrollo vía WIFI similar a los dispositivos Arduino orientado a Internet de las cosas, posee su propio lenguaje, enviando comandos mediante el puerto serial (CP2102). integra un potente microcontrolador con arquitectura de 32 bits (más potente que el Arduino Due) y conectividad Wi-Fi. El SoM(System on Module) ESP-12E fabricado por Ai-Thinker integra en un módulo el SoC ESP8266, memoria FLASH, cristal oscilador y antena WiFi en PCB.

Soporta el desarrollo de aplicaciones en diferentes lenguajes como: Arduino, Lua, MicroPython, C/C++, Scratch. Se puede usar un lenguaje de programación conocido y hacer uso de un IDE de manera practica y de disponer de toda la información sobre proyectos y librerías disponibles en la red. Los Community Social de Arduino es muy activa y da fiabilidad a toda la plataforma. Haciendo uso de todos los top en base a las placas de desarrollo.

1.4. Formulación del Problema.

¿Qué protocolos de comunicación en la capa aplicación de internet de las cosas será el más adecuado para la transferencia de datos aplicado en un sensor de color?

1.5. Justificación e importancia del estudio.

La Actual investigación está dentro de una de las Líneas de Investigación aprobadas por la Universidad Señor de Sipán (USS) en tecnologías de la información (genéricamente hablar) con que protocolos de comunicación se trabaja en la capa de aplicación en el internet de las cosas evidenciando con cual se debe optar a trabajar y cuál es el que está en un top para desarrollar y validar sus datos

Los protocolos de la capa aplicación se muestran dentro de los últimos años a través del Internet de las cosas es de valioso punto el tema de investigación ya que el valor de la investigación se aprecia más sabiendo que hay poco recurso sin embargo se gestiona el valor de la investigación para poder desarrollar Algunos modelos para determinar una mejoría en el campo de investigación. En la Investigación Comparar Protocolos de, comunicación en la capa transporte es significativo. Al terminar el proyecto se podrá brindar que protocolo es el que tiene un mejor rendimiento y realizar pruebas de ellos analizando comparaciones de acuerdo con las variables ya investigadas durante el periodo. En el I Desarrollo del presente Proyecto de investigación se utilizará un sensor de color y un dispositivo Wifi o Modulo ESP32 ya que es de fácil acceso a la utilización de la tecnología, es económicamente viable para no hacer una gran inversión en nuestro trabajo.

1.4. Hipótesis.

El protocolo de comunicación MQTT de la capa aplicación es la más eficientes para la

Transferencia de Datos en sensor de color, ya que en el anexo N° 1

nos muestra que están por encima de otros protocolos evidenciando su velocidad de Transferencia, retardo y disminución de pérdida de paquetes.

1.5. Objetivos.

1.5.8. Objetivo general.

1.5.8.1. Evaluar protocolos de internet de las cosas para la Transferencia de Datos aplicado a un sensor de color.

1.5.9. Objetivos específicos.

1.5.9.1. Seleccionar protocolos de la capa aplicación de Internet de las Cosas

1.5.9.2. Diseñar la Arquitectura de la red para ejecutar la transferencia de datos

1.5.9.3. Implementar los Protocolos de comunicación.

1.5.9.4. Evaluar los Protocolos de comunicación.

2. MATERIAL Y MÉTODO

2.3. Tipo y Diseño de Investigación.

2.3.8. Cuantitativa Tecnológico Aplicada:

Es cuantitativa porque estudia la relación entre las variables de la Investigación que han sido cuantificadas para luego procesarlas y analizarlas estadísticamente y sus indicadores están representados en cantidades numéricas (Domínguez, 2007).

2.3.9. DISEÑO DE INVESTIGACIÓN

2.3.9.1. Investigación Diseño Cuasi Experimental:

Se justifica de esa manera porque los cambios surgen desde la variable teniendo en cuenta la realidad al probar la existencia de una relación causal entre dos o más variables.

2.4. Población y muestra.

Se realizó un estudio teórico de 7 protocolos de comunicación de la capa aplicación utilizada en diferentes proyectos de investigación relacionada al tema que se detallan en la Tabla 10.

La muestra se ha determinado por conveniencia se ha considerado 2 protocolos de comunicación de la capa aplicación: MQTT y HTTP de acuerdo con el análisis de investigación y debido al rendimiento y han generado porcentajes con buen rendimiento en comparación con la demás población quedando de la siguiente manera y poder brindar una investigación diferente con la finalidad de que el estudio sea de buen aporte para otras investigaciones que se muestran en la referencia del Anexo N° 1.

2.5. Variables, Operacionalización.

Tabla 7. Variables, Operacionalización

Variables	Dimensión	Indicador	Ítem	Técnica e instrumentos de recolección de datos
VARIABLE DEPENDIENTE	Velocidad	Velocidad de Transferencia	$T = T_m / AB$	
VARIABLE INDEPENDIENTE	Tiempo	Tiempo de Retardo	$Retardo = T_r - T_s$	Observación Documentación Bitácora Wireshark
VARIABLE INDEPENDIENTE	Megabytes	Pérdida de Paquetes	$Paquetes_Perdidos = \sum Paquetes_Enviados - \sum Paquetes_Recibidos$	

5.3.1. Variables independientes:

- Protocolos de Comunicación de Internet de las Cosas

5.3.2. Variable dependiente:

- Transferencia de Datos

2.6. Técnicas e instrumentos de recolección de datos, validez y confiabilidad.

Tabla 8. Técnicas e instrumentos

Nombre	Descripción
BW	Máximo ancho de banda teórico del "enlace más lento" entre el host de origen y el host objetivo (medido en bits/segundo)
P	Tasa de transferencia real en el momento de la transferencia (medida en bits por segundo)
T	Tiempo en que se debe producir la transferencia de archivos (medido en segundos)
S	Tamaño de archivo en bits

Velocidad de Transferencia

Siempre es medido en bits por segundos y esta puede depender directamente de los medios, tipos de conexión física, límites de cachés

Transferencia = tamaño del archivo/ancho de banda ($T = T_m / AB$)

Pérdida de Paquetes

Se define como la diferencia del número total de paquetes enviados por el remitente y el número total de paquetes recibidos en el receptor

$$\text{Paquetes_Perdidos} = \sum \text{Paquetes_Enviados} - \sum \text{Paquetes_Recibidos}$$

Tiempo de Retardo

Se define como el intervalo que experimentan los paquetes cuando viajan a través de varias redes de un nodo a otro. $Retardo = T_r - T_s$ T_r está transmitiendo el tiempo del paquete específico, mientras que T_s es el tiempo de recepción del paquete.

Wireshark

Programa creado con software libre y se utiliza para analizar paquetes de datos en una red para mostrar problemas y análisis de rendimientos en cuanto a tramas de paquetes, herramienta para el análisis de protocolos y permite examinar datos o de un archivo de captura recuperados en un disco. Se puede analizar la información capturada, a través de los detalles y sumarios por cada paquete. Wireshark incluye un completo lenguaje para filtrar lo que queremos ver y la habilidad de mostrar el flujo reconstruido de una sesión de TCP y su uso es muy portable ya que se puede utilizar en diferentes sistemas operativos.

Documentación

Hace referencia a una disciplina académica que ha sido objeto de análisis lingüísticos se podría definir como la ciencia del procesamiento de información, que proporciona un compendio de datos con un fin determinado, (Luis Fernando Ramos Simón, 2003. Introducción a la administración de la información, Universidad Complutense de Madrid: Síntesis).

Observación

La observación consiste en saber seleccionar aquello que queremos analizar. Para la observación lo primero es plantear previamente qué es lo que interesa observar. En definitiva, haber seleccionado un objetivo claro de observación.

Bitácora

La bitácora es un instrumento de recolección de datos que acompaña al observador de campo y tiene la función de guardar de forma primaria y así como se presentan todos los datos que se consideran pertinente en el tema de investigación el cual

representa ser un cuaderno o diario donde se registran con la mayor fidelidad posible lo que se observa de la realidad, también puede agregarse apreciaciones del observador ,emociones y reacciones ,el propósito real del registro detallado es tener una fuente de datos de un sector de una realidad en un momento determinado. De Nikolaus (2001).

2.7. Procedimiento de análisis de datos.

FIABILIDAD: En esta investigación se cumplirá con expectativas y su implementación.

VALIDACIÓN: Se validarán los instrumentos de recolección de datos que han obtenido por el estudio.

CREDIBILIDAD: Se mostrará resultados del estudio sean verdaderos por los investigadores que lo realizan y participan en el estudio.

2.8. Criterios éticos.

Se toma en cuenta las referencias bibliográficas y los pensamientos, definiciones y frases técnicas de los autores para que sean utilizadas en el proyecto utilizando la técnica del parafraseo, teniendo en cuenta no caer en la duplicidad o plagio en la investigación.

DERECHOS DE AUTOR:

Toda la información utilizada en la construcción de la tesis debidamente citada con sus autores y referencias asignadas.

2.9. Criterios de Rigor Científico.

La calidad de esta investigación dependió del rigor con la que fue realizada, eso condiciona su credibilidad. Los criterios de rigor científico empleados en esta investigación son descriptos en la siguiente tabla.

Tabla 9. Criterios de Rigor Científico

Criterios	Características éticas del criterio
Fiabilidad	El estudio para la selección de protocolos nos permitirá analizar qué tan fiable son los protocolos utilizados.
Consistencia	Los datos recolectados para este trabajo son de carácter científico y formal. El análisis realizado a los datos está hecho con total profesionalidad aplicando habilidades y conocimientos de la ingeniería y de la investigación para mantener la consistencia de los datos y resulte en información fidedigna y útil
Validez	Los datos obtenidos de la recopilación de datos por el sensor de color y en su implementación serán correctamente evaluados haciendo validez de las variables a medir para lograr dar un resultado válido que verifique la hipótesis
Originalidad	Se citará las fuentes bibliográficas de la información, con el fin de demostrar la inexistencia del plagio

3. RESULTADOS.

3.3. Resultados en Tablas y Figuras.

Figura 12. Pruebas de Velocidad de Transferencia de Protocolo MQTT

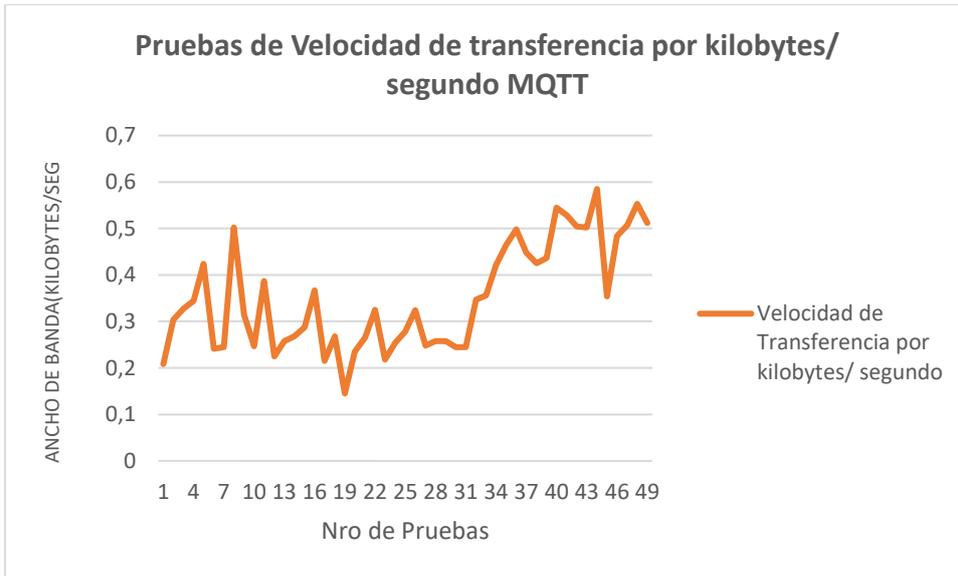


Figura 13. Pruebas de Velocidad de Transferencia de Protocolo HTTP

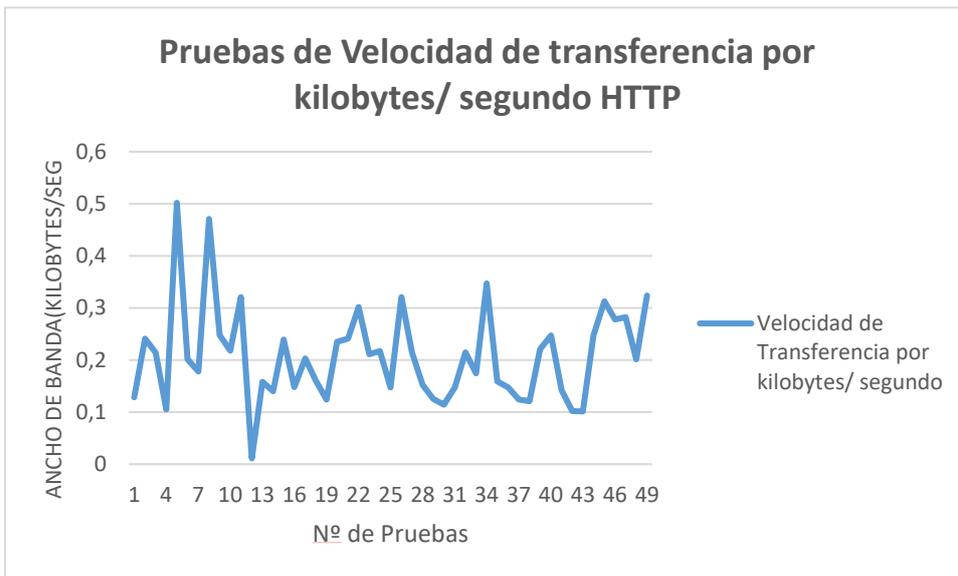


Figura 14. Pruebas de Pérdida de Paquetes del Protocolo MQTT

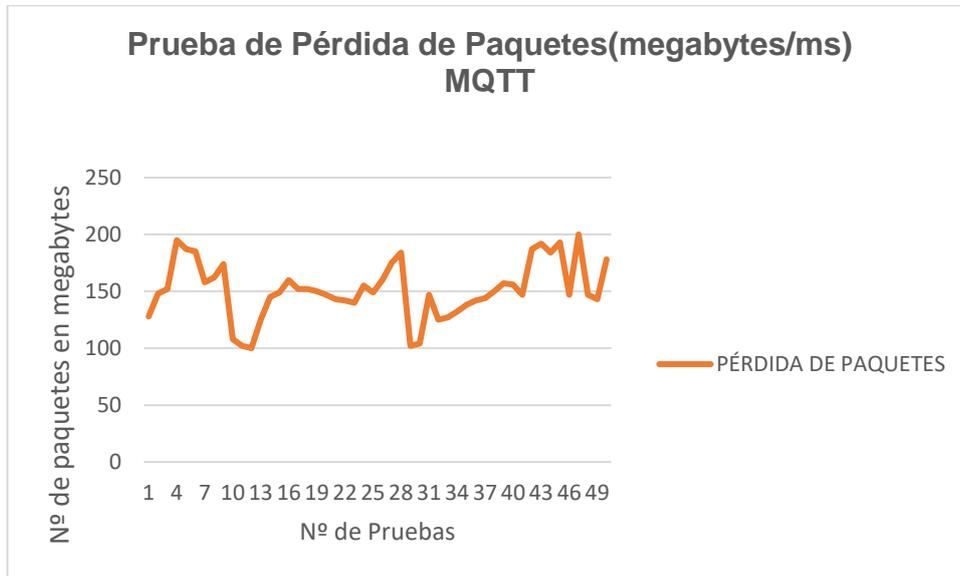


Figura 15. Pruebas de Pérdida de Paquetes del Protocolo HTTP

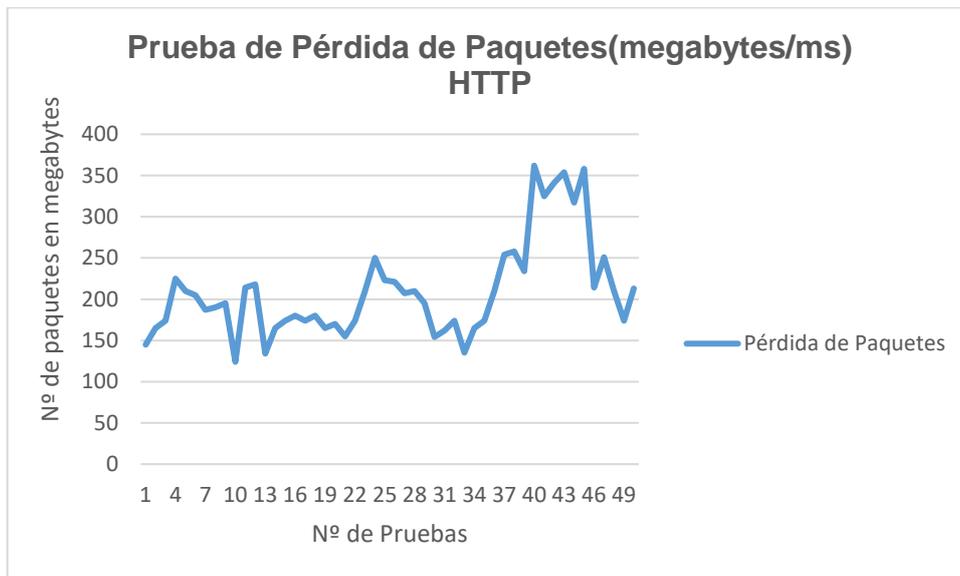


Figura 16. Pruebas de Tiempo de Retardo del Protocolo MQTT

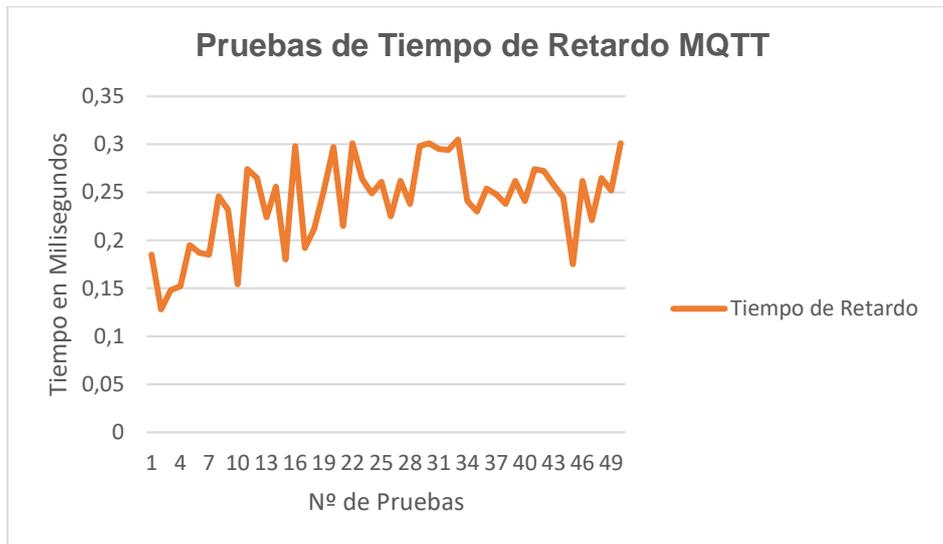
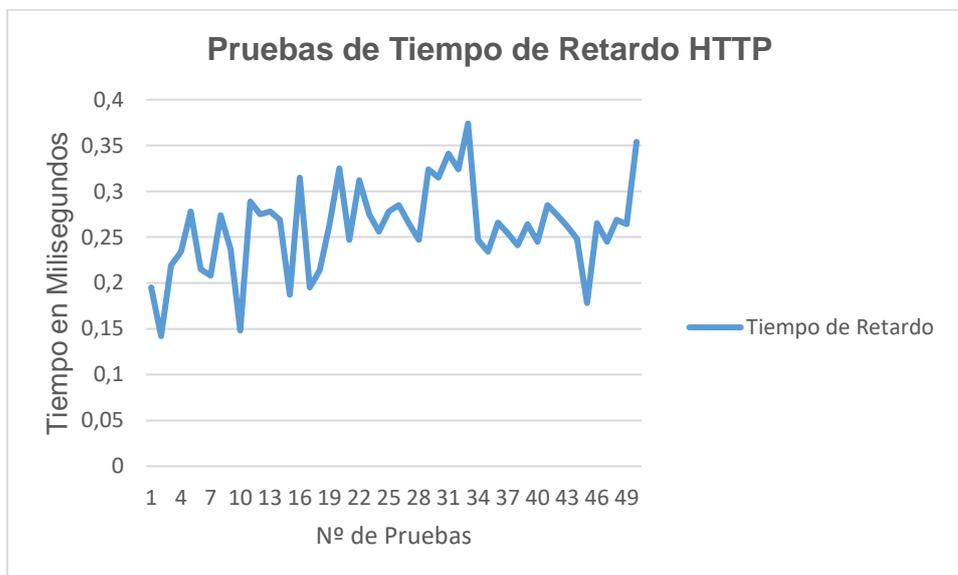


Figura 17. Pruebas de Tiempo de Retardo del Protocolo HTTP



3.4. Discusión de resultados.

3.5. Aporte práctico.

Método:

El método que se utilizó en el trabajo de investigación es propio del área de tecnologías de información que usa el método de la observación con su

instrumento de fichas de observación. Así mismo las fases del método se detallan en el siguiente gráfico.

Se realizó la selección de protocolos de comunicación de la capa aplicación de internet de las cosas (IOT), al evidenciar diferentes protocolos se compararon 7 protocolos para poder determinar cuál es el que tiene mejores principios y alcanza un mejor nivel de Transferencia de acuerdo a las variables que se mide en las investigaciones, siendo así Evaluados para enviar datos desde un procesador de imagen (sensor de color) para ser conectado a un dispositivo Arduino luego de la lectura que se realice estos datos son enviados mediante un dispositivo ESP8266 Wifi a un servidor gratuito en ese escenario realizarse las pruebas de acuerdo a los tres Indicadores se realice la evaluación y determinar los resultados pertinentes.

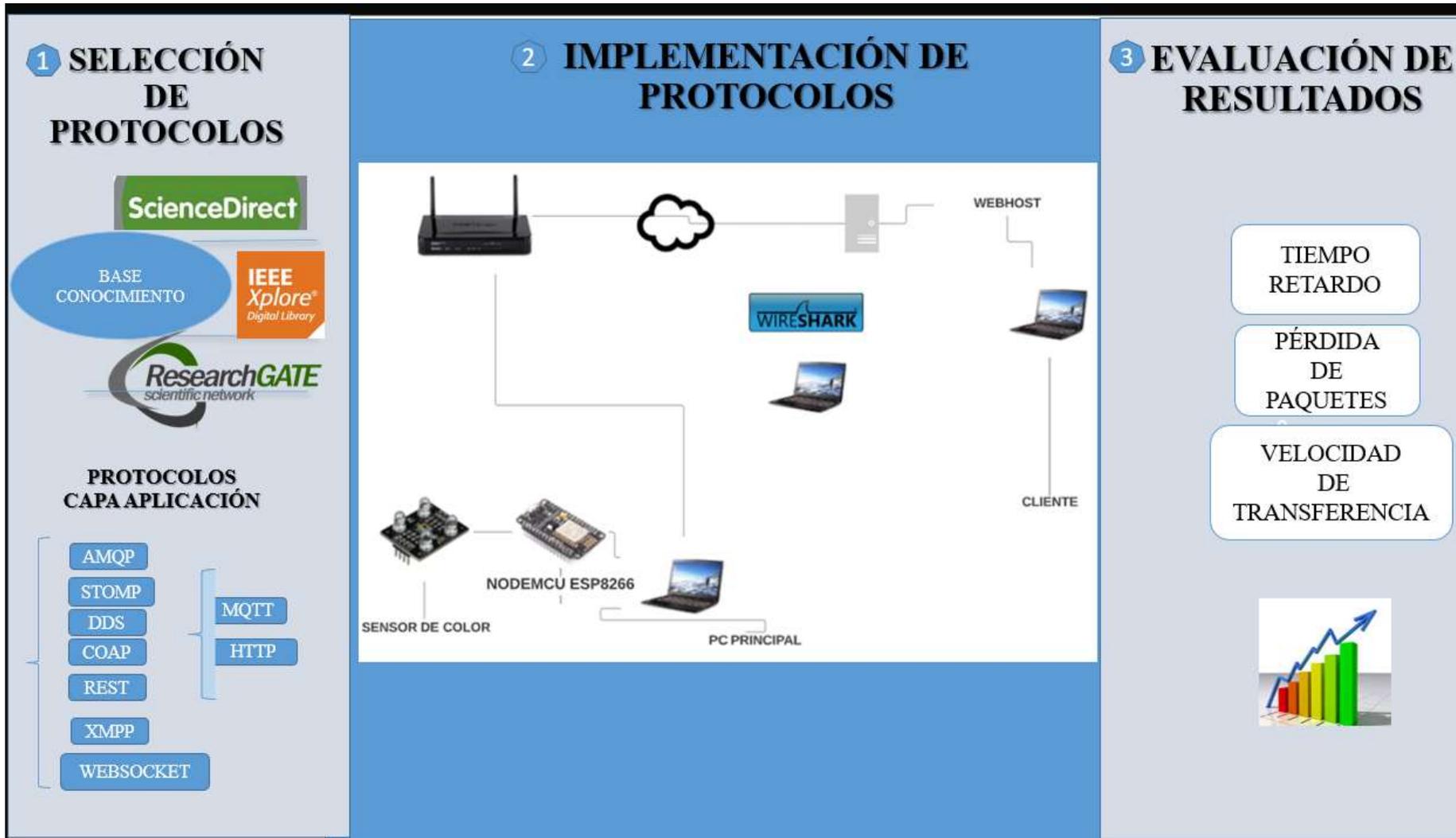


Figura 18. Método Propuesto. Fuente: Elaboracion Propia

Tabla 10. Selección de Protocolos de Comunicación Capa Aplicación

AUTORES	PROTOCOLOS CAPA APLICACIÓN IOT
IETF (Internet Engineering Task Force,2001)	CoAP
(JP Morgan Chase, 2004).	AMQP
(Dr. Andy Stanford-Clark de IBM y Arlen Nipper de Arcom,1999)	MQTT
(Empresas Real-Time Innovations1 - Thales Group2 ,2003)	HTTP
(Jeremie Miller,2005)	XMPP
Grupo de Trabajo de Ingeniería de Internet (IETF,1998)	STOMP
Grupo de Trabajo de Ingeniería de Internet (IETF,2010)	WEBSOCKET

Fuente Elaboración Propia con Referencia a (Ravulabaru,2018)

CoAP utilizado para abarcar redes de baja potencia y pequeñas cantidades en cuanto a memoria RAM.

MQTT: desarrollado e implementado por software libre y se utiliza para entornos restringidos y redes de menor consumo de banda ancha.

AMQP permite que las aplicaciones del usuario interactúan directamente con el cliente haciendo uso de elementos que están dentro de un entorno de envío de datos continuo con el servidor, soporta transacciones.

XMPP solo diseñado para envíos de mensajería instantánea

DDS Protocolo de tipo publicación/suscripción concebida para sistemas de tiempo-real. Es un estándar abierto y descentralizado.

WEBSOCKET fue diseñado para funcionar bien con la infraestructura web que ya existe, la especificación del protocolo define que la conexión Websocket comienza su vida como una conexión http, garantizando una compatibilidad total con el mundo pre-websocket.

STOMP basado en cadena de textos, emplea HTTP y mensajes de texto para buscar el máximo de interoperabilidad.

1. Selección de protocolos de la capa transporte de Internet de las cosas.

Se realizó la selección de protocolos de comunicación de la capa aplicación de una base de conocimiento en donde diferentes investigaciones fueron tomadas para poder ser evaluadas por conveniencia y de acuerdo con las variables se realizó la evaluación en donde se evidencio cual tendría un mejor rendimiento y demostrar cual es el mejor para generar una mejor transferencia de datos a nivel de una arquitectura de red. (Juese,2018) y (Polese, 2018).

Tabla 11. Criterios de la Evaluación

CRITERIOS DE VALUACION	
NIVEL DE IMPACTO	PUNTAJE
BUENO	5
REGULAR	3
MALO	1

Fuente: Elaboración Propia

El método de la selección de los protocolos es la siguiente

Los criterios que se toman en la Tabla N° 11 equivalen a que si el protocolo cumple con los requisitos aceptables en donde se evaluara siendo el puntaje 5 si es bueno,3 si es regular y 1 si es malo.

C1: Velocidad de Transferencia

C2: Pérdida de Paquetes

C3: Retardo de Extremo a Extremo

*Tabla 12. Evaluación y resultados de los protocolos de Internet de las Cosas
Capa Aplicación (IOT)*

Nº	PROTOCOLOS	CRITERIO-1	CRITERIO-2	CRITERIO-3	RESULTADOS
		Velocidad de Transferencia	Pérdida de Paquetes	Tiempo de Retardo	
1	MQTT	3	3	5	11
2	HTTP	3	3	5	11
3	COAP	5	3	3	11
4	HTTP	3	3	5	11
5	AMQP	5	3	3	11
6	XMPP	3	3	1	7
7	DDS	2	3	1	6

Fuente Elaboración Propia referenciando Enterprise Internet of things Handbook, (Ravulabaru, 2018)

De todos los protocolos encontrados en la matriz con variables se analizaron y concluimos que los mejores protocolos para la transferencia de datos serían los 2 mostrando en la tabla N° 10 mostrada a continuación.

2. Diseño de la Arquitectura de la red para ejecutar la transferencia de datos

Se muestra un escenario de la arquitectura de la red con las siguientes características

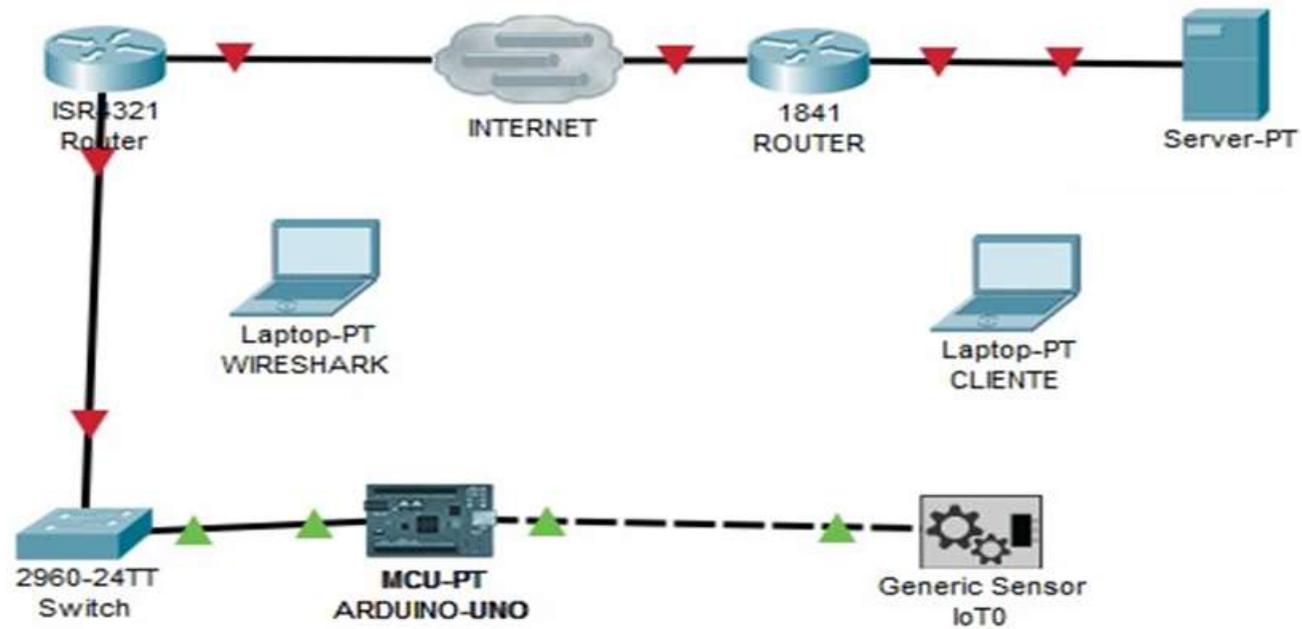


Figura 19. Diseño Físico de la Red.

Fuente: Entorno Packetracer

En la Figura se observa el diseño físico de la red, se utilizó el sensor de color con la finalidad de que se capturen los datos aleatoriamente teniendo en cuenta el tiempo, velocidad y porcentaje de luz que muestra al capturar los datos cuando se muestran en consola, el sensor de color requiere tener un voltaje no mayor a 5 voltios, con el motivo de no generar ningún corto circuito en los dispositivos, se requiere estar conectado a un dispositivo Arduino para que se capturen los datos y se puedan ver reflejados desde su Entorno de desarrollo es decir desde la consola de Arduino es necesario que el sensor de color este calibrado para que los colores sean directamente captura

Se realizó la configuración de un Router para las pruebas necesarias y mantener acceso a internet con la única función de poder enviar datos, asimismo se mantiene una red de área local (LAN) para pruebas locales, la velocidad de internet depende mucho del proveedor que no impide en cierto sentido aprovechar los recursos compartidos ya que no se va a medir la velocidad del servicio de conectividad si no los Indicadores en gestión a medir cada protocolo. Se utilizó un Hosting gratuito con la finalidad de que se realicen los envíos de los datos y se reciban los mensajes teniendo la función de un bróker principal en el diseño mostrado, en principio MQTT tiene por finalidad un Bróker teniendo un Suscriptor y un Publicador mediante ese escenario se realiza el análisis de datos; también es necesario que los puertos y los servicios estén configurados correctamente para que no se evidencie algún tipo de error al realizar las pruebas funcionales, Mqtt tiene en su estructura ya implementada en los servicios de seguridad que viene predeterminados por ello hacen más fácil el consumo y producción de recursos al ser evaluados. Se tomó en cuenta utilizar Wireshark con la finalidad que, se realicen las pruebas funcionales y evalúen las variables o indicadores para cada protocolo y mostrar los resultados favorables para cada uno; se tomó en cuenta en el diseño de la red el cliente quien tiene la función de poder tener acceso a los datos enviados(publicador) del bróker del cual es parte fundamental la recepción por ellos también se toma en cuenta analizar el tráfico con Wireshark, para los protocolos de comunicación

de la capa aplicación es necesario descargar las librerías e implementarlas, haciendo uso de la captura de datos que se van a mostrar al momento que sean enviados para ser publicados y suscritos aplicado a la arquitectura de la red mostrada; para una mayor comprensión se ha ilustrado la Figura N°20 el diseño lógico de la red mostrado en un cuadro pictográfico.

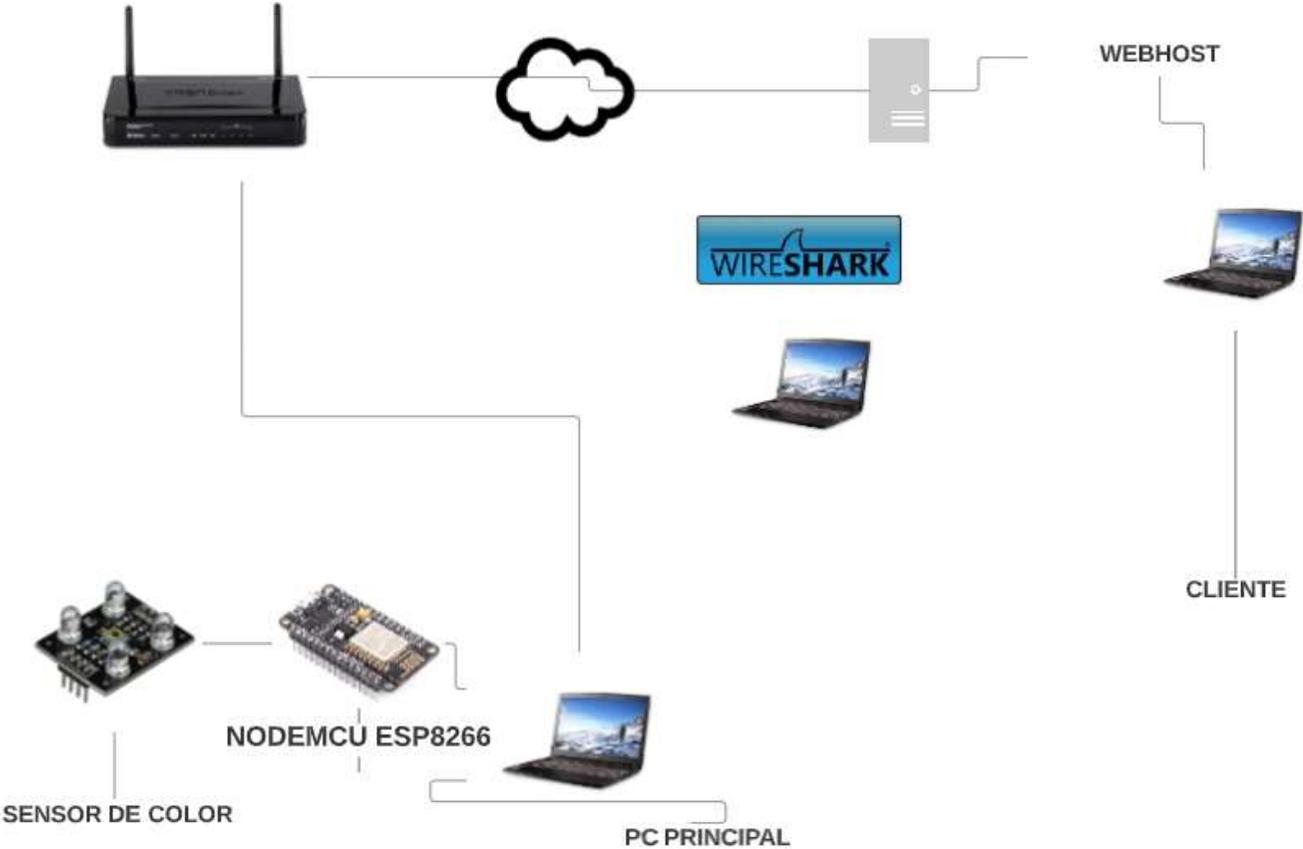


Figura 20. Descripción Pictográfico de la Arquitectura de la Red.

Fuente: Entorno LucidChart

En el siguiente cuadro pictográfico se evidencia la tipología Estrella, para ello se define parámetros de ubicación para mostrar un mejor enfoque, en la parte del servidor se requiere que los ip sean configurados de manera que se muestren los datos, dirigidos al servidor.

Siguiendo el procedimiento de poder diseñar la arquitectura de red, explicaremos como se ejecuta nuestro procedimiento.

El primer paso inicia tomando al sensor de color el cual está compuesto por 64 lodos que leen 16 colores por cada uno, estos 4 led capturan los colores RGB tienen una capacidad de resistencia de 5 Voltios como máximo, el sensor de color necesariamente es interconectado de manera física por hilos de dispositivos jumpers para equilibrar el voltaje y ello está normalizado dentro de la arquitectura de red; el sensor con el dispositivo Arduino Uno en segunda instancia la data recibida será transferidos hacia el equipo Arduino Uno mediante un conector R3 ; se programara e implementara cada protocolo de Internet de las cosas como MQTT y HTTP al terminar este proceso se mostraran los resultados midiendo las variables.

3. Implementación de Protocolos de comunicación en la capa aplicación.

La implementación se realizará utilizando dos protocolos de comunicación de la capa aplicación MQTT y HTTP con el lenguaje de programación C++ Compilado en el IDE de Arduino, se descargaron previamente las librerías de cada Protocolo para poder gestionar la ejecución de cada uno, en principio cada unidad de data recibida será implementada para determinar su utilización y quien mejor realiza un desempeño optimo con él envió de datos evaluando cada indicador.

Implementación de Sensor de Color

En el siguiente código se muestra la estructura en lenguaje C++ compilado en el IDE Arduino, con la finalidad de que los datos capturados por el sensor de color sean mostrados de acuerdo con el color detectado; es necesario que el sensor de color mientras las pruebas se realicen estén conectados a la placa Arduino Uno

utilizada, de tal manera se realizó la configuración necesaria evidenciándose de manera correcta la utilización del sensor.

//se declaran las constantes para cada entrada entre el sensor de color y de la placa arduino

```
const int s0 = 1;
```

```
const int s1 = 2;
```

```
const int s2 = 3;
```

```
const int s3 = 4;
```

```
const int out = 5;
```

```
Int rojo = 0;
```

```
Int verde = 0;
```

```
Int azul = 0;
```

```
void setup (){
```

```
    Serial.begin (9600);
```

```
    pinMode (s0, OUTPUT);
```

```
    pinMode (s1, OUTPUT);
```

```
    pinMode (s2,OUTPUT);
```

```
    pinMode (s3, OUTPUT);
```

```
    pinMode (out, INPUT);
```

```
    digitalWrite (s0, HIGH);
```

```
    digitalWrite (s1, HIGH);
```

```
}
```

```
void loop (){
```

```
// Cada color capturado por el sensor se muestra en un valor hexadecimal derivado de un valor serial.
```

```
    Color ();
```

```
    Serial.print (" ");
```

```
    Serial.print (rojo, DEC);
```

```
    Serial.print (" ");
```

```
    Serial.print (verde, DEC);
```

```

Serial.print (" ");
Serial.print (azul, DEC);

if (rojo < azul && verde > azul && rojo < 35)
{
  Serial.println (" Rojo");
}
else if (azul < rojo && azul < verde && verde < rojo)
{
  Serial.println (" Azul");
}

else if (rojo > verde && azul > verde )
{
  Serial.println (" Verde");
}
else{
  Serial.println(" ");
}
delay(900);
}

void color()
{
  DigitalWrite (s2, LOW);
  DigitalWrite (s3, LOW);
  rojo = pulseIn(out, digitalRead(out) == HIGH ? LOW : HIGH);
  DigitalWrite (s3, HIGH);
  azul = pulseIn(out, digitalRead(out) == HIGH ? LOW : HIGH);
  digitalWrite(s2, HIGH);
  verde = pulseIn(out, digitalRead(out) == HIGH ? LOW: HIGH);
}

```

Información de acceso

Corredor:	broker.emqx.io
Puerto TCP:	1883
Puerto de socket web:	8083
Puerto TCP/TLS:	8883
Puerto websocket/TLS:	8084
Autoridad certificada:	broker.emqx.io-ca.crt ↓

El brókerMQTT 5 que se utiliza es Gratuito y Estándar y se ajusta a los requerimientos que se solicita dentro de la implementación del Protocolo MQTT. Se ha utilizado el Código de Arduino para la Estructura del Protocolo MQTT y generar las pruebas necesarias en base a el envío de los datos acorde al Método Propuesto.

La estructura del empieza desde que se descargan las librerías a utilizar, seleccionando la placa con la que se tiene que Trabajar y agrupando las Entradas y Salidas del Sensor donde se Reciben los datos para luego que estas mismas sean enviadas a la web, el sensor de color contiene la variación de Colores RGB las cuales contiene 3 parámetros las cuales varían y se identifican al momento que se generan las solicitudes.

PROYECTO_HTTP_FER

```
#include <WiFi.h>
#include <HTTPClient.h>

#define S0 17
#define S1 18
#define S2 32
#define S3 25
#define salidaSensor 33

const char *ssid = "Fam_Chucua";//Nombre de tu red
const char *password = "469817341506";//Clave de tu red

String serverName = "https://swampy-fittange.000webhostapp.com/update.php";

int R = 0;
int G = 0;
int B = 0;

String RE = "0";
String GE = "0";
String BE = "0";

String color;

void setup() {
  Serial.begin(115200);

  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(salidaSensor, INPUT);

  WiFi.begin(ssid, password);
  Serial.println("Connecting");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
```

Figura 21. Estructura de Programación del Protocolo HTTP en Arduino

```

PROYECTO_MQTT_FER §

#define B0 27
#define B1 18
#define B2 32
#define B3 28
#define salidaSensor 33

const char *mqtt_server = "broker.emqx.io";
const int mqtt_port = 1883;

WiFiClient espClient;
PubSubClient client(espClient);

const char *ssid = "Fam_Chucas";//Nombre de tu red
const char *pass = "489817941504";//Clave de tu red

int B = 0;
int S = 0;
int R = 0;

String BE = "0";
String GE = "0";
String RE = "0";

String color;

#define MQTT_COLORES "proyectoFernando/Colores12345"

void(* Resetear) (void) = 0;//Como se palazamos el boton B3

void setup_wifi() {
  int cont = 0;
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, pass);
  Serial.println("Conectando a la red: " + String(ssid));
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    cont++;
    if (cont == 20) {
      *****
    }
  }
}

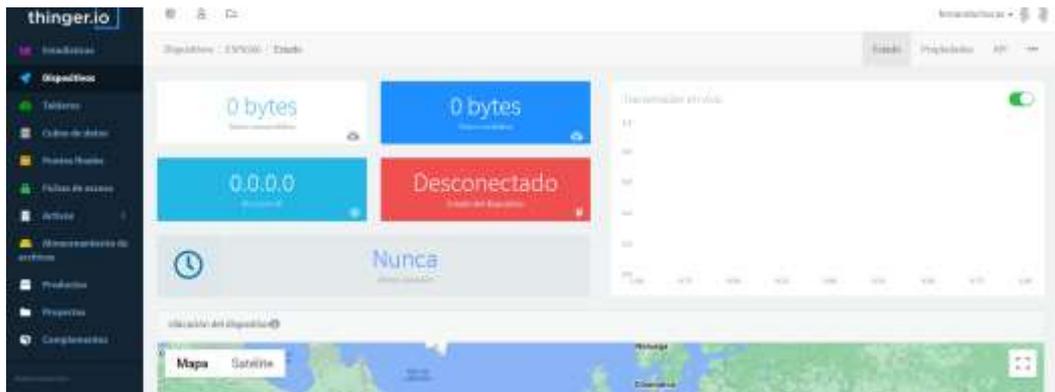
```

Figura 22. Estructura de Programación del Protocolo MQTT en Arduino

Figura 23. Se realiza la creacion del WebHost para la Pruebas y Envio de Datos



Thinger.io es una plataforma de código abierto, diseñado para que sea accesible y de manera pública y distribuida de manera pública, en estos 3 últimos años esta plataforma se ha distribuido en 182 países con más de 50 mil usuarios entre las empresas grandes que las utilizan son Electrolux, Prosegur o Google; Permite configurar, almacenar y realizar monitorización de Datos



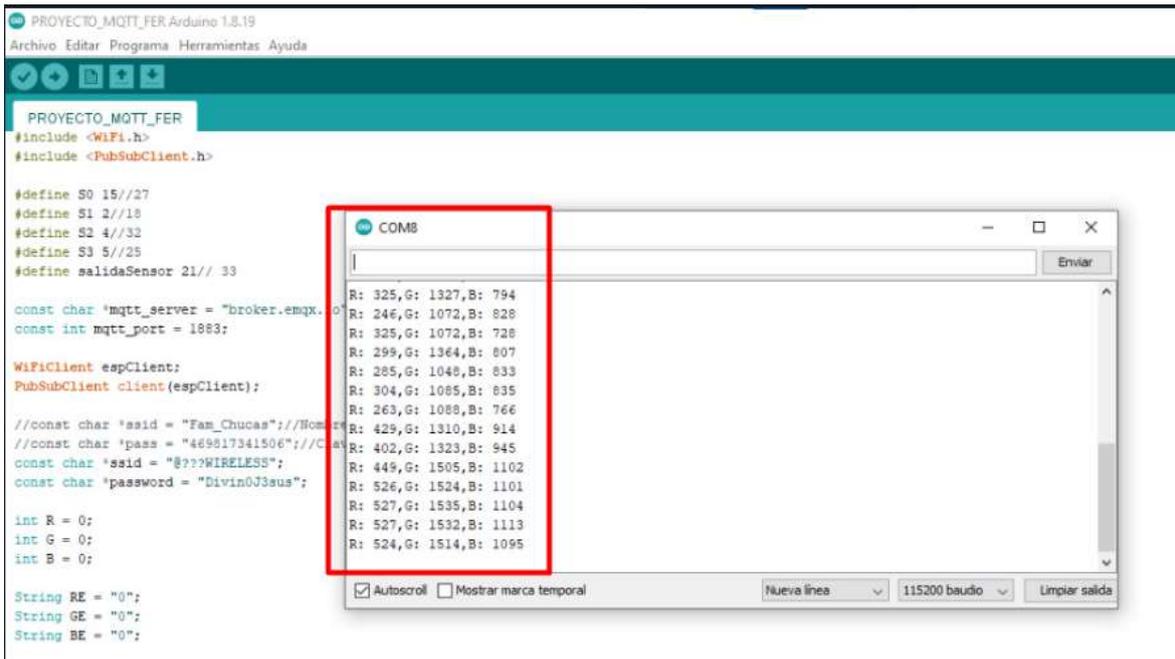


Figura 24. Ejecución donde muestra la Lectura de los Colores mediante el Patrón de Salida de los Valores tomado por el Sensor de Color.

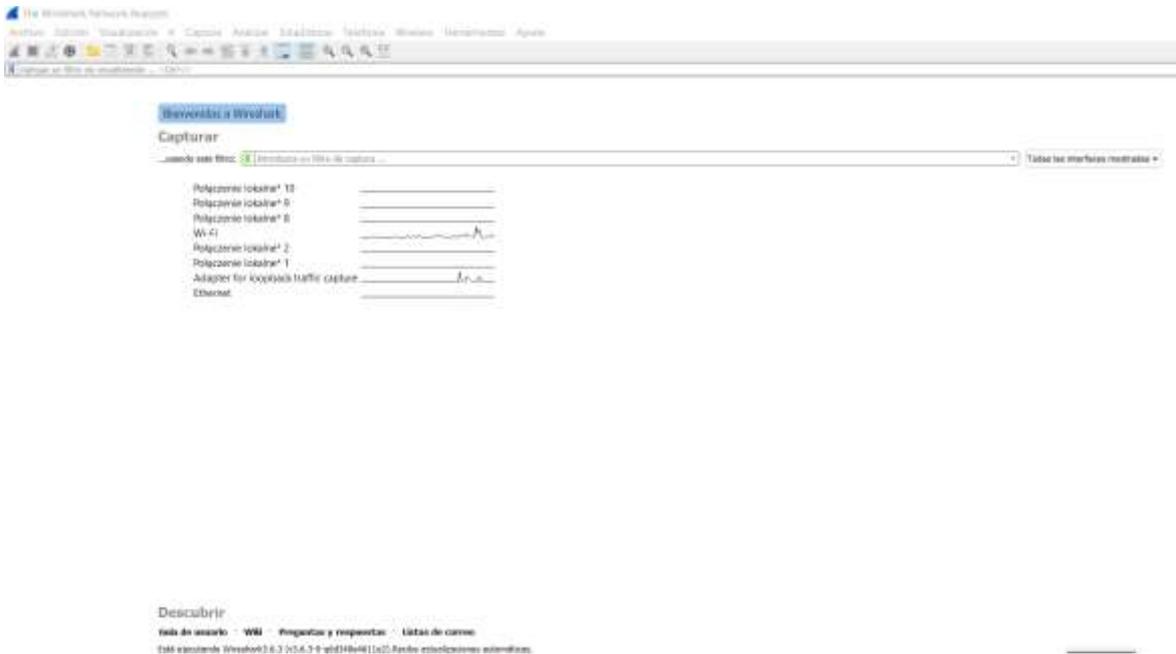


Figura 25. Se realiza la Configuración del Programa Wireshark

En esta aplicación se analiza el tráfico de la Red y se muestran los Paquetes por cada capa

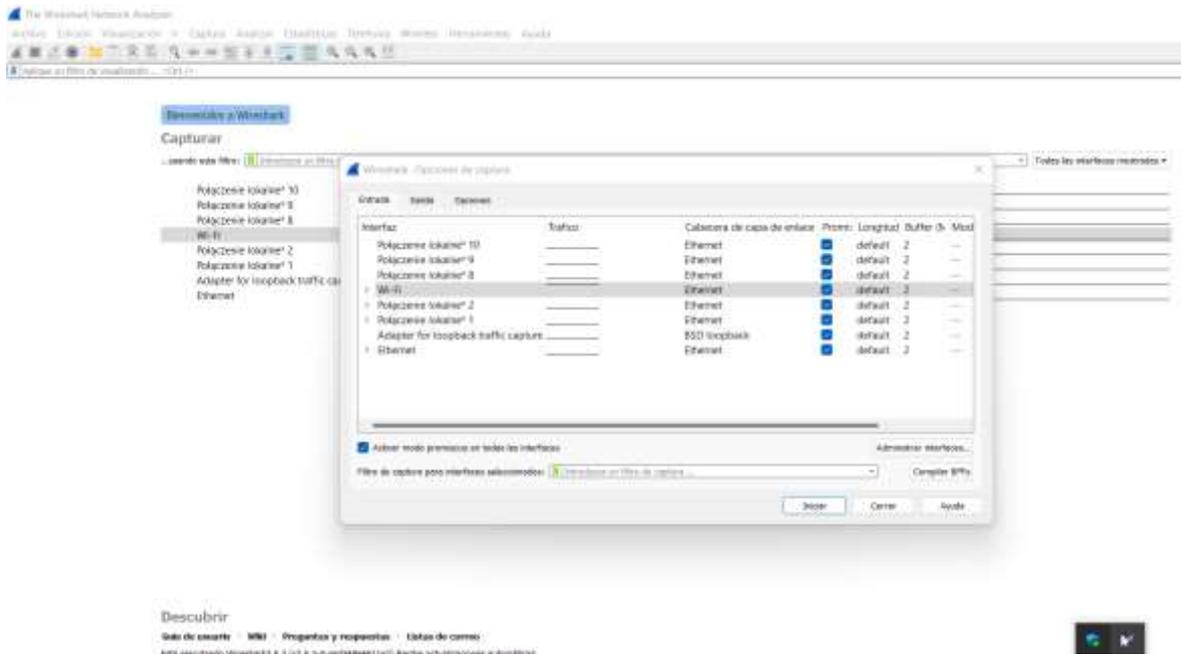


Figura 26. Se selecciona el Dispositivo Wifi del cual se tomará los Datos del Trafico.

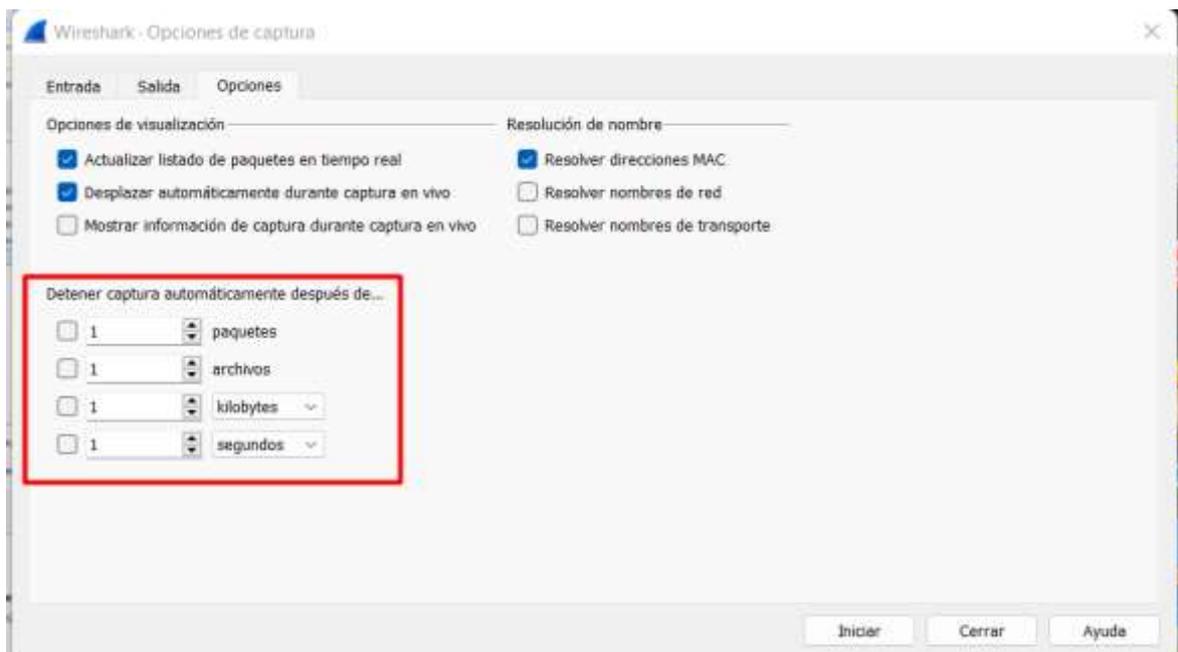


Figura 27. Se realiza la Toma y Configuración de la cantidad de paquetes, archivos, kilobytes o segundos a tomar el tráfico de Datos.

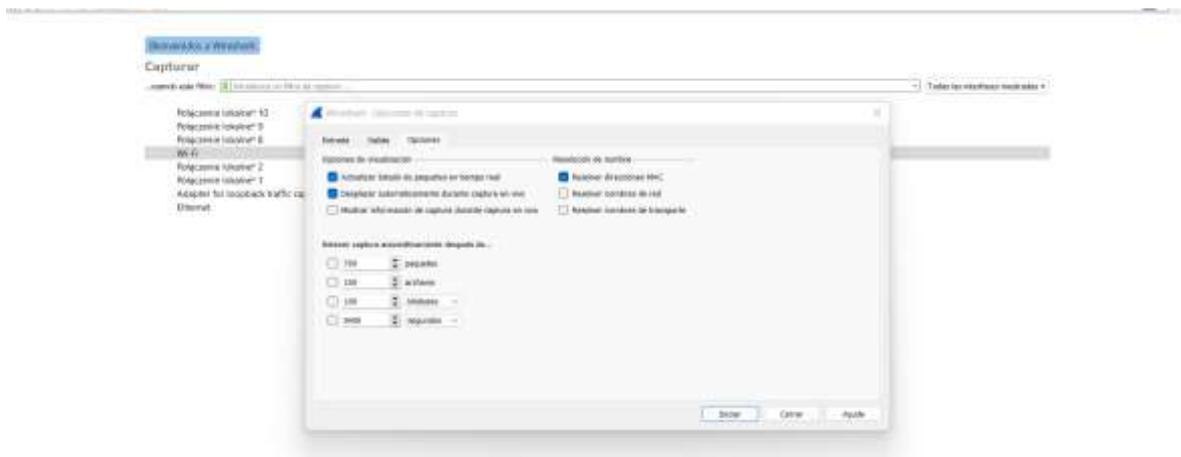


Figura 28. Se realiza la Configuración para Obtener los Datos y Paquetes a Capturar.

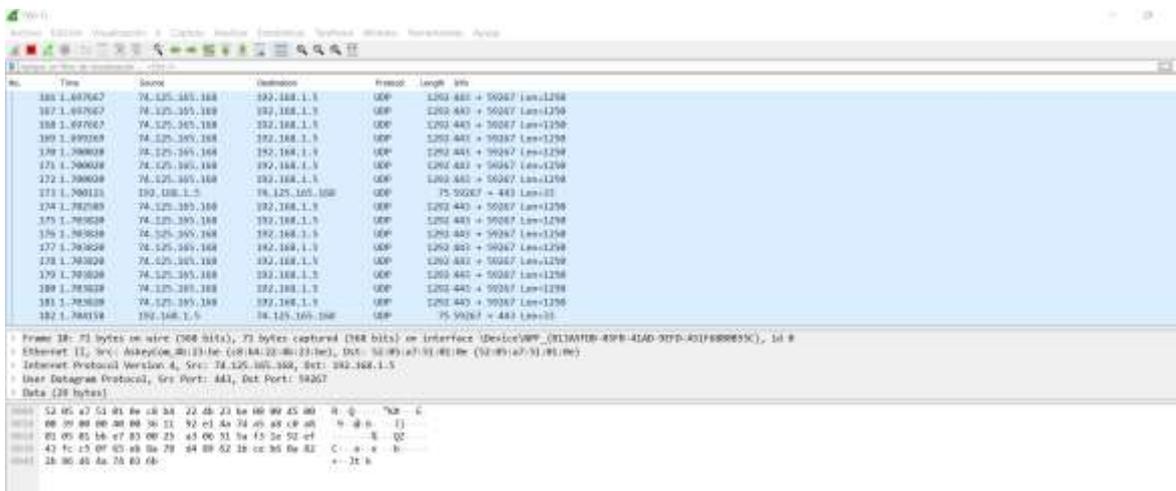


Figura 29. Al iniciar el Wireshark ya tenemos Trafico no solo del envio de Datos, sino también la maquina ya genera paquetes de acuerdo con el Análisis de Red que se tiene en el campo de Pruebas

4. CONCLUSIONES Y RECOMENDACIONES

4.3. Conclusiones.

- La Evaluación se realizó en un ambiente no controlado para monitorear los resultados que se hicieron al detectar desde un sensor de color se realiza la conexión con el Dispositivo esp8266, siguiendo con el procedimiento de envío de datos para evaluar en cada escenario a Mqtt y a Http ,los resultados al evaluar los protocolos de comunicación de la capa aplicación concluyen que el protocolo Mqtt tiene un tiempo de velocidad de transferencia por encima de 0.553 Mb en un porcentaje de 95% frente al protocolo Http con una velocidad de transferencia de 0.471 Mb/s con un porcentaje de 95%; Mqtt tiene una pérdida de paquetes de 0.215 Mb/s en un porcentaje de 95% frente al protocolo Http con una pérdida de paquetes al 95 % de 0.358 megabytes; Mqtt tiene un tiempo de Retardo de envío de paquetes de 0.264 ms con un porcentaje válido del 95% frente al protocolo Http con un porcentaje del 95% válido de 0.374 ms. si se tiene un tiempo optimo debe de ser no mayor a 300ms
- Se concluye además que para realizar los Estudios con la capa aplicación del protocolo de comunicación no hay ninguno Estándar o prototipo a seguir y esto hace que el proceso de aprendizaje sea más beneficioso en evaluaciones futuras y mejoras de procesos simultáneos.

4.4. Recomendaciones.

- Es recomendable que las nuevas tecnologías permitan un mejor acceso de aprendizaje para así obtener mayor impacto en investigación y se propongan más proyectos en base a las tecnologías emergentes.
- En base a mi tesis se pretende ser punto de investigación a favor de más estudiantes y generar más empatía con las nuevas tendencias tecnológicas como ya se encuentran realizando Machine Learning en su rama con Inteligencia Artificial.

REFERENCIAS.

- RESEARCHGATE. (2018).de RESEARCHGATE:
https://www.researchgate.net/publication/328189515_A_Survey_on_Recent_Advances_in_Transport_Layer_Protocols.
- Gartner. (2018). *Says the Internet of Things Installed Base Will grow*.
- IEEE. (2001). *RADIO FREQUENCY IDENTIFICATION AND THE ELECTRONIC PRODUCTCODE*. New York: IEEE.
- Jesus Antonio Vega Uribe, M. A. (2006). *Transformaciones ineaales y no Lineales para espacios de Color en procesamiento de imágenes*. *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*. 18.
- Jhon Ramirez, C. P. (2017). *Performance analysis of communication protocols*. *Performance analysis of communication protocols*, 7.
- Jorge Salazar, S. S. (2017). *Internet de las Cosas*. *TECH*, 34.
- Michele Polese, F. C. (2018). *(A Survey on Recent Advances*. *ResearchGate*, 19.
- Pérez, S. C. (2017). *Dispositivos y Protocolos en Redes LAN Y WAN*. *ResearchGate*, 242.
- Ramos, H. (2013). *Redes Inalambricas de Sensores Inteligentes Aplicacion a la Monitorizacion de Variables Fisiologicas* . España: JDARE.
- Ravulavaru, A. (2018). *Enterprise Internet of Things Handbook*. UK: Pack Publishing.
- Wheeb, A. H. (2017). *Evaluación del desempeño de UDP, DCCP, SCTP y TFRC para Diferente flujo de tráfico en redes cableadas*. *SCIENCIDIRECT*.
- Erkin Kirdan, Marc-Oliver Pahl, Georg Carle(2021). *Open Source Mqtt Evaluation- Technical University of Munich, IEEE*
- N. Naik. *Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP*. In *2017 IEEE International Systems Engineering Symposium (ISSE)*, pages 1–7, Oct 2017. <https://ieeexplore.ieee.org/abstract/document/8088251>.

Muhammad Yaqoob Junejo. (2018). On the use of Multipath Transmission using SCTP. ResearchGate, 6.

Pérez, S. C. (2017). Dispositivos y Protocolos en Redes LAN Y WAN. ResearchGate, 242.

RESEARCHGATE. (2018). Obtenido de RESEARCHGATE: (https://www.researchgate.net/publication/328189515_A_Survey_on_Recent_Advances_in_Transport_Layer_Protocols)

SADOUNI Salheddine, 1. B.-L. (2017). SCTP-MANET NEW EXTENSION OF SCTP PROTOCOL FOR THE OPTIMIZATION OF MANET. ResearchGate, 14.

Stallings, W. (2009). Data and Computer Communications. Usa: Pearson Education.

Wheeb, 2. (2017). medir-color-arduino-colorimetro-tcs3200. www.luisllamas.es, <https://howtomechatronics.com/tutorials/arduino/arduino-color-sensing-tutorial-tcs230-tcs3200-color-sensor/><https://www.luisllamas.es/medir-color-arduino-colorimetro-tcs3200>.

Wheeb, A. H. (2017). Evaluación del desempeño de UDP, DCCP, SCTP y TFRC para Diferente flujo de tráfico en redes cableadas.

J. Granjal, E. Monteiro, and J. Sa Silva, "End-to-end transport-layer security for internet-integrated sensing applications with mutual and delegated ecc public-key authentication," in IFIP Networking Conference, 2013, May 2013, pp. 1–9.

management, T. d. (2017). Tiempo de management. Obtenido de <https://www.tiempodemanagement.com/ponentes/kevin-ashton-futurismo/>

Martínez, J.-M., Mejía, J., Muñoz, M., & al., e. (2017). La Seguridad en Internet de las Cosas:Analizando el Tráfico de Información en Aplicaciones para iOS. Revista Electrónica de Computación ,Informática,Biomédica y Eelectrónica, Vo 6,num 1.

Neven Nikolov. (2020). Investigación de MQTT, CoAP, HTTP y XMPP IOT Protocolos de comunicación para Sistemas Integrados. Department of Computer Systemc and Technology, Technical University of Sofia,Bulgaria.IEEE

Yufei An, F. Richard Yu, Fellow, Jianqiang Li, Jianyong Chen y Victor C.M. Leung, (2020,") Borde de Inteligencia (EI): marco de detección de anomalías HTTP habilitado para Internet de las cosas (IoT)".IEEE(1-17)

Sandy, S.Prayogo.,Yulisdin M,Bayu K. Yakti.(2020)."El uso y rendimiento de MQTT y CoAP como protocolo de aplicación de Internet de las cosas utilizando NodeMCU

ESP8266". Indonesia Universidad de Jakarta. IEEE Xplore(1-5).

Esteban D. Volentini, Carlos Albaca Paravàn, Josè Younes, Sergio D. Saade , Luis A. Tek Maria de los A. Gòmez Lòpez.(2020), " Descripción de un sistema IoT para la medición y registro de la calidad del aire", Argentina Universidad Nacional de Tucuman, IEEE(1-7)

Melvin Bender, Erkin Kirdan, Marc-Oliver Pahl, Georg Carle, (2021), " Evaluación de MQTT de código abierto". Universidad Técnica de Múnich-Alemania. IEEE (1-4).

Wolfgang Bziuk, Cao Vien Phung, Jasenka Dizdarevic y Admela Jukan,(2018). "Sobre el rendimiento de HTTP en aplicaciones IoT:Un análisis de latencia y rendimiento". Universidad Técnica de Braunschweig, Alemania, IEEE(1-6).

Yufei An, F. Richard Yu, Fellow, Jianqiang Li, Jianyong Chen y Victor C.M. Leung, (2020) "Borde de Inteligencia (EI): marco de detección de anomalías HTTP habilitado para Internet de las cosas (IoT)". IEEE(1-17)

Olumide Kayode , Ali Saman Tosun(2019) "Análisis de tráfico de IoT utilizando HTTP Proxy". Universidad Zilina, Republica Eslovaca, IEEE(1-4).

Abdur Rahman, Shanto Roy, M. Shamim Kaiser y Shahidul Islam,(2018). "Un marco S-MQTT ligero de varios niveles para asegurar la comunicación entre nodos de IoT de gama baja", Universidad Verde de Bangladesh, Dhaka. IEEE(1-5).

Hafsiya T.H Binet Rose y Kodakara ,(2021) ,"Un dispositivo portátil de monitoreo de salud basado en la nube de IoT", Departamento de ingeniería Electrónica y de Comunicaciones. de Sahrdaya, India. IEEE(1-3).

Padlan Alqinsi, Nanang Ismail, Ian Joseph Matheus Edward y Wahyudin Darmalaksana,(2018), " IoT-Based UPS Monitoring System Using MQTT Protocols", Indonesia. IEEE (1-4).

ANEXOS.

Anexo 1. Resolución de aprobación del proyecto de investigación



FACULTAD DE INGENIERÍA, ARQUITECTURA Y URBANISMO

RESOLUCIÓN N° 1142-2018/FIAU-USS

Chiclayo, 14 de diciembre de 2018

VISTO:

El Dictamen de Aprobación de Proyecto de Tesis N° 037-2018/FIAU-IS-USS de fecha 14 de diciembre de 2018, para la ejecución de la Tesis titulada: *"PROTOCOLOS DE COMUNICACIÓN DE INTERNET DE LAS COSAS PARA LA TRANSFERENCIA DE DATOS APLICADOS A UN SENSOR DE COLOR"*, presentada por el(los) estudiante(s) **CHUCAS REQUEJO JOSE FERNANDO** de la Escuela Académico Profesional de **INGENIERÍA DE SISTEMAS** y;

CONSIDERANDO:

Que, de conformidad con la Ley Universitaria N° 30220 en su artículo 48° que a letra dice: *"La investigación constituye una función esencial y obligatoria de la universidad, que la fomenta y realiza, respondiendo a través de la producción de conocimiento y desarrollo de tecnologías a las necesidades de la sociedad, con especial énfasis en la realidad nacional. Los docentes, estudiantes y graduados participan en la actividad investigadora en su propia institución o en redes de investigación nacional o internacional, creadas por las instituciones universitarias públicas o privadas."*;

Estando a lo expuesto, y en uso de las atribuciones conferidas y de conformidad con las normas y reglamentos vigentes;

SE RESUELVE:

ARTÍCULO 1°: APROBAR, el Proyecto de Tesis denominado *"PROTOCOLOS DE COMUNICACIÓN DE INTERNET DE LAS COSAS PARA LA TRANSFERENCIA DE DATOS APLICADOS A UN SENSOR DE COLOR"*, perteneciente a la Línea de Investigación **TECNOLOGÍAS DE LA INFORMACIÓN - INTERNET DE LAS COSAS**, a cargo del(los) estudiante(s) **CHUCAS REQUEJO JOSE FERNANDO**, de la Escuela Académico Profesional de **INGENIERÍA DE SISTEMAS**.

ARTÍCULO 2°: ESTABLECER, que la inscripción de la Tesis se realice a partir de emitida la presente resolución, y tendrá una vigencia máxima de 02 años.

REGÍSTRESE, COMUNÍQUESE Y ARCHÍVESE

Cc: Dirección de Investigación, CPGYT, Interesadas, Archivo

ADMISIÓN E INFORMES

074 481610 - 074 481632

CAMPUS USS

Km. 5, carretera a Pimentel

Chiclayo, Perú

www.uss.edu.pe

Anexo 2. MATRÍZ DE CONSISTENCIA LÓGICA DE TRABAJO DE INVESTIGACIÓN

MATRÍZ DE CONSISTENCIA LÓGICA DE TRABAJO DE INVESTIGACIÓN							
Enfoque metodológico							
Título	PROTOSCOLOS DE COMUNICACIÓN DE INTERNET DE LAS COSAS PARA LA TRANSFERENCIA DE DATOS APLICADOS A UN SENSOR DE COLOR						
Tipo de investigación	Problema	Variabes	Indicadores	Población	Muestra	Método de recolección de Datos	Técnicas de procesamiento de datos
Cuamthiva Tecnológico Aplicada	¿Qué protocolo de comunicación en la capa transporte de internet de las cosas será el más adecuado para la transferencia de datos aplicados en un sensor de color?	Variable Dependiente: Protocolos de Comunicación de Internet de las Cosas Variable Independiente: Transferencia de Datos	Velocidad de Transferencia Tiempo de Retardo Pérdida de Paquetes	Se realizó un estudio teórico de 7 protocolos de comunicación de la capa aplicación utilizados en diferentes proyectos de investigación relacionados al tema que se detallan en el anexo "1".	Se realizó un estudio teórico de 7 protocolos de comunicación de la capa aplicación utilizados en diferentes proyectos de investigación relacionados al tema.	Observación	WinVisshark
Diseño de investigación	Hipótesis	Objetivo General	Objetivos específicos	Método propuesto y desarrollado	Resultados preliminares		
Investigación Diseño Cuasi Experimental	El protocolo de comunicación MQTT de la capa aplicación es la más eficiente para la Transferencia de Datos en sensor de color, ya que en el anexo "1" se muestra que están por encima de otros protocolos evaluando su velocidad de transferencia, retardo y disminución de pérdida de paquetes.	Evaluar protocolos de internet de las cosas para la Transferencia de Datos aplicados a un sensor de color.	<ol style="list-style-type: none"> 1. Seleccionar protocolos de la capa aplicación de internet de las cosas. 2. Diseñar la Arquitectura de la red para evaluar la transferencia de datos. 3. Implementar los Protocolos de comunicación. 4. Evaluar los Protocolos de comunicación. 				