



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL**

CARRERA DE INGENIERÍA ELECTRÓNICA

**TRABAJO DE TITULACIÓN PREVIO A LA
OBTENCIÓN DEL TÍTULO DE INGENIERO ELECTRÓNICO**

PROYECTO TÉCNICO:

**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO IOT PARA
EL MONITOREO DE PARÁMETROS AMBIENTALES
APLICADOS A LA AVICULTURA PARA LA CRIANZA DE
POLLOS DE GRANJA UTILIZANDO HARDWARE DE BAJO
COSTO Y AWS (AMAZON WEB SERVICES).**

AUTORES:

**BRYAN ISRAEL GARCÍA VACA
FÉLIX ROBERTO MORA CRUZ**

TUTOR:

MSC. KLEVER CARRIÓN

GUAYAQUIL – ECUADOR

2021

Certificado de responsabilidad y autoría del trabajo de titulación

Nosotros, Bryan Israel García Vaca y Félix Roberto Mora Cruz, estudiantes de ingeniería electrónica de la Universidad Politécnica Salesiana certificamos que los conceptos desarrollados, análisis realizados y las conclusiones del presente proyecto de titulación son de exclusiva responsabilidad de los autores y es propiedad intelectual de la Universidad Politécnica Salesiana.

Guayaquil, julio del 2021



(f) _____
BRYAN ISRAEL GARCÍA VACA
C.I. 0932208564



(f) _____
FÉLIX ROBERTO MORA CRUZ
C.I. 0921455564

Certificado de cesión de derechos de autores

Nosotros, Bryan Israel García Vaca y con documento de identificación N.º 0932208564 y Félix Roberto Mora Cruz con documento de identificación N.º 0921455564, manifestamos nuestra voluntad de ceder a la Universidad Politécnica Salesiana los derechos patrimoniales en calidad de autores del proyecto de titulación titulado “DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO IOT PARA EL MONITOREO DE PARÁMETROS AMBIENTALES APLICADOS A LA AVICULTURA PARA LA CRIANZA DE POLLOS DE GRANJA UTILIZANDO HARDWARE DE BAJO COSTO Y AWS (AMAZON WEB SERVICES)”, mismo que ha sido desarrollado para optar por el título de ingeniero electrónico con énfasis en telecomunicaciones, en la Universidad Politécnica Salesiana, quedando la universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la ley de propiedad intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada.

En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, julio del 2021



(f) _____
BRYAN ISRAEL GARCÍA VACA
C.I. 0932208564



(f) _____
FÉLIX ROBERTO MORA CRUZ
C.I. 0921455564

Certificado de dirección del trabajo de titulación

Por medio de la presente doy a conocer que el proyecto de titulación “DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO IOT PARA EL MONITOREO DE PARÁMETROS AMBIENTALES APLICADOS A LA AVICULTURA PARA LA CRIANZA DE POLLOS DE GRANJA UTILIZANDO HARDWARE DE BAJO COSTO Y AWS (AMAZON WEB SERVICE)”, presentado por Bryan Israel García Vaca y Félix Roberto Mora Cruz para optar por el título de ingeniero electrónico con énfasis telecomunicaciones, se ajusta a las normas establecidas por la Universidad Politécnica Salesiana, por tanto, autorizo su presentación ante las autoridades pertinentes.

Guayaquil, julio del 2021



(f) _____
MSC. KLEVER CARRIÓN
DIRECTOR DE PROYECTO DE TESIS

Dedicatoria

Dedico esta tesis primeramente a Dios por su sabiduría y amor infinitos, a mis padres por su apoyo constante en mis estudios, a mi novia Elizabeth quien fue siempre uno de mis pilares y quien me alentó a seguir cuando parecía que no iba a finalizar.

A mis profesores de la especialidad de telecomunicaciones quienes me inspiraron y fueron el ejemplo a donde quería llegar con la carrera.

Para todos ellos hago esta dedicatoria quienes supieron marcar mi camino y dejar su huella permanente.

BRYAN ISRAEL GARCÍA VACA

Dedico mi tesis en primer lugar a Dios, que me ha dado salud y fortaleza para culminar esta gran etapa de mi vida. A mis padres que han sido pilares fundamentales a lo largo de esta carrera; gracias a su apoyo, su paciencia, su amor y su entrega día a día, por motivarme a seguir adelante hasta alcanzar la meta. A mi esposa por ser mi compañera en cada momento apoyándome y dándome su apoyo incondicionalmente.

FÉLIX ROBERTO MORA CRUZ

Resumen del proyecto

Año	Alumnos	Tutor de Proyecto de titulación	Proyecto de titulación
2021	Bryan Israel García Vaca Félix Roberto Mora Cruz	MSc. Klever Carrión	Diseño e implementación de un prototipo IOT para el monitoreo de parámetros ambientales aplicados a la avicultura para la crianza de pollos de granja utilizando hardware de bajo costo y AWS (AMAZON WEB SERVICES)

Las granjas avícolas en zonas rurales de la costa ecuatoriana por lo general no usan tecnificación para la crianza de pollos para el consumo masivo, dando como resultado alta mortalidad en las aves durante la crianza, por tal motivo se plantea en este trabajo de investigación un diseño e implementación de un prototipo IoT (Internet de las cosas) para el monitoreo de parámetros ambientales aplicados a la avicultura para la crianza de pollos de granja utilizando hardware de bajo costo como ESP32 y servidores web en la nube con AWS (Amazon Web Services) para el monitoreo y almacenamiento de información ambiental como humedad de suelo, humedad ambiental, temperatura e intensidad lumínica que permita tomar decisiones y mejorar las condiciones técnicas para la crianza de aves de granja.

Las tecnologías del internet de las cosas aplicadas a la avicultura ayudarán a los pequeños avicultores para una correcta tecnificación en la crianza de pollos de granja y de consumo. En este contexto se plantea un prototipo IoT que mediante sensores y actuadores podrán controlar la temperatura y humedad de los galpones de pollos, con el control de la luminosidad dentro de los galpones de pollos, se permitirá una correcta temperatura para la crianza de estas aves.

Este trabajo de investigación plantea un prototipo para aplicarse en una maqueta de galpón de pollos que podría replicarse para grandes galpones y ponerla en práctica en las pequeñas granjas avícolas. Con el control y monitorización de los parámetros ambientales en los galpones de pollos se podrá tomar decisiones correctas para su mejor crianza y desarrollo. La metodología aplicada en este trabajo de investigación es descriptiva y exploratoria ya que se analizarán los valores obtenidos por los sensores de los factores ambientales para una toma de decisiones en la crianza de pollos.

Como resultados obtenidos de la investigación se aplican tecnologías en la nube como AWS para subir la información de los sensores cada cierto tiempo y poder monitorizar temperatura, humedad del ambiente, temperatura de suelo, intensidad lumínica. Mediante actuadores se encenderán ventiladores que permitan bajar la temperatura y controlar la cantidad de luz dentro del criadero de pollos.

Abstract

Year	Students	Degree Project Tutor	Technical Degree Project
2021	Bryan Israel García Vaca Félix Roberto Mora Cruz	MSc. Klever Carrión	Design and implementation of an IOT prototype for monitoring environmental parameters applied to poultry farming for raising farm chickens using low-cost hardware and AWS (AMAZON WEB SERVICES)

Poultry farms in rural areas of the Ecuadorian coast generally do not use chicken breeding technology for mass consumption, resulting in high mortality in birds during breeding, so a design and implementation of an IoT (Internet of Things) prototype for monitoring environmental parameters applied to poultry for farm chicken rearing using low-cost hardware is proposed in this research work such as ESP32 and cloud web servers with AWS (Amazon Web Services) for monitoring and storing environmental information such as soil moisture, ambient humidity, temperature, and light intensity to make decisions and improve technical conditions for farm bird rearing.

Internet of Things technologies applied to poultry will help small poultry for proper technification in farm and consumer chicken rearing. In this context, an IoT prototype is proposed that through sensors and actuators will be able to control the temperature and humidity of chicken sheds, with the control of the luminosity within the chicken sheds, a correct temperature will be allowed for the rearing of these birds.

This research work proposes a prototype to be applied to a chicken shed model that could be replicated for large sheds and put into practice on small poultry farms. With the control and monitoring of environmental parameters in chicken sheds, correct decisions can be made for better rearing and development. The methodology applied in this research work is descriptive and exploratory as the values obtained by the sensors of environmental factors for decision-making in chicken rearing will be analyzed.

As results obtained from the research, cloud technologies such as AWS are applied to upload sensor information from time to time and to be able to monitor temperature, ambient humidity, soil temperature, light intensity. Fans will be turned on using actuators to lower the temperature and control the amount of light inside the chicken hatchery.

Índice general

Certificado de responsabilidad y autoría del trabajo de titulación	ii
Certificado de cesión de derechos de autores	iii
Certificado de dirección del trabajo de titulación.....	iv
Dedicatoria	v
Resumen del proyecto	vi
Abstract	vii
Índice general.....	viii
Índice de figuras.....	xi
Índice de tablas	xiv
Introducción.....	xv
1. El problema	1
1.1. Descripción del problema.....	1
1.2. Antecedentes.....	1
1.3. Importancia y alcance	2
1.4. Delimitación	2
1.4.1. Delimitación temporal	2
1.4.2. Delimitación espacial	3
1.4.3. Delimitación académica	3
1.5. Explicación del problema	3
1.6. Objetivos.....	4
1.6.1. Objetivo general	4
1.6.2. Objetivos específicos	4
2. Fundamentos teóricos	5
2.1. Avicultura	5
2.2. Características de los pollos de engorde.....	6
2.2.1. Alimentación del pollo de engorde	6
2.2.2. Anatomía del pollo de engorde.....	7
2.2.3. Características de una buena calidad de pollito	8
2.3. Sectores en el Ecuador adecuados para la avicultura	9
2.4. Factores que influyen en la crianza de pollitos de granja	12
2.4.1. Temperatura y rendimiento del pollo	14
2.4.2. Humedad relativa	17
2.4.3. Ventilación.....	18
2.4.4. Iluminación durante la crianza	19
2.4.5. Conservación de energía.....	20
2.5. Avicultura y el Internet de las cosas	21
2.5.1. Granja avícola inteligente: solución integrada que utiliza una red basada en WSN y GPRS.....	22
2.5.2. Granja inteligente con red de sensores inalámbricos	22
2.5.3. Granja avícola inteligente basada en Internet.....	23
2.5.4. Incorporar tecnología de detección inteligente en la industria avícola.....	23
2.5.5. Diseño e implementación de relé inteligente basado en monitoreo remoto de amoníaco en granjas avícolas.....	23
2.5.6. Uso del Raspberry Pi y Arduino para el monitoreo inteligente de la granja	23
2.5.7. IoT (Internet de las Cosas)	24

2.5.8.	Su Funcionalidad	25
2.6.	Sensores y actuadores para IoT	26
2.7.	Tipos de Sensores	26
2.7.1.	Sensor DHT11	26
2.7.2.	Sensor Humedad de suelo	28
2.7.3.	Sensor intensidad de luz (luxómetro)	29
2.8.	Actuadores	30
2.9.	Tipos de Actuadores	31
2.9.1.	Actuador Hidráulico.....	31
2.9.2.	Características	31
2.9.3.	Actuador Neumático	31
2.9.4.	Características	32
2.9.5.	Actuador Eléctrico	32
2.10.	Actuadores básicos de IoT	33
2.10.1.	Servomotores	33
2.10.2.	Motores paso a paso.....	34
2.10.3.	Motores de CC (motores de rotación continua).....	34
2.10.4.	Actuador lineal.....	35
2.10.5.	Relevo.....	36
2.10.6.	Selenoide.....	36
2.11.	ESP32	37
2.11.1.	Características Técnicas del ESP32.....	38
2.12.	Servicio web en la nube	39
2.12.1.	Beneficios de la nube	40
2.12.2.	Potencial de la nube	40
2.12.3.	Hardware	40
2.12.4.	Importancia de la nube.....	41
2.13.	AWS (Amazon Web Services)	41
2.13.1.	¿Para qué sirve Amazon Web Services?	42
2.13.2.	Beneficios	42
2.13.3.	Herramientas/Servicios	42
2.13.4.	Herramientas Administrativas	43
2.14.	AWS IoT.....	43
2.14.1.	¿Qué tipos de servicios ofrece AWS IoT?.....	43
2.14.2.	Casos de uso de la AWS IoT	43
2.15.	TimeStream	44
2.15.1.	Beneficios	44
2.16.	Amazon DynamoDB	45
2.16.1.	Beneficios	45
2.17.	Descripción de las configuraciones en AWS	46
2.17.1.	Protocolo MQTT	47
2.17.2.	Reglas de AWS IoT	47
2.17.3.	Temas MQTT	47
2.17.4.	Nombres de temas	48
2.17.5.	Filtros de temas	48
2.18.	Grafana Cloud.....	48
2.18.1.	Características.....	49
2.18.2.	Integraciones en la nube	49
2.18.3.	Paneles de control en la nube (Grafana).....	50
3.	Marco metodológico.....	51
3.1.	Tipo de investigación.....	51

3.2.	Metodología del trabajo.....	51
3.2.1.	Control y monitoreo de un criadero avícola controlado por microcontrolador desde un sitio web dinámico.	51
3.2.2.	Control y monitorización de una granja avícola	52
3.2.3.	Diseño e implementación de sistema de monitoreo automatizado en granja avícola.....	52
3.3.	Título de la propuesta.....	53
3.4.	Justificación	53
3.5.	Beneficiarios de la propuesta.....	53
3.6.	Propuesta de solución.....	53
3.7.	Diseño electrónico del prototipo IoT	55
4.	Resultados	67
4.1.	Montaje de sensores en maqueta.....	67
4.2.	Parámetros de temperatura ambiente y humedad relativa.	74
4.3.	Configuraciones en AWS	75
4.4.	Demostración del proceso de registros de los datos.	77
4.5.	Creación de reglas para AWS IoT	77
4.6.	Descripción de la configuración en Grafana.....	87
5.	Análisis de resultados	91
5.1.	Monitor de sensores desde IDE de Arduino	91
5.2.	Pruebas de envío de mensajes GSM	92
5.3.	Monitor de sensores desde AWS	93
5.4.	Visualización de datos en Grafana Cloud.....	94
6.	Conclusiones.....	98
7.	Recomendaciones	99
	Bibliografía	100
	Anexos	102
	Guía de práctica de laboratorio	102
	Monitoreo de sensores en IDE de Arduino	104
	Monitor de sensores desde AWS	105
	Código de ESP32 en IDE Arduino.....	108

Índice de figuras

Figura 2.1 Granja de pollos	5
Figura 2.2 Pollo de engorde	7
Figura 2.3 Cría del Pollo de Engorde	8
Figura 2.4 Sectores en el Ecuador adecuados para la avicultura	9
Figura 2.5 Tecnificación en la Avicultura.....	11
Figura 2.6 Tipos de aves criadas en el Ecuador	12
Figura 2.7 Pollitos en galpón	14
Figura 2.8 Galpón de pollos.....	16
Figura 2.9 Niveles de amoniaco	18
Figura 2.10 Iluminación en galpones de pollos	20
Figura 2.11 IoT aplicada a la avicultura.....	22
Figura 2.12 Avicultura y el Internet de las cosas	24
Figura 2.13 (Internet de las Cosas)	25
Figura 2.14 Sensor DTH11	27
Figura 2.15 Sensor humedad de suelo	28
Figura 2.16 Sensor luxómetro	30
Figura 2.17 Actuador hidráulico.....	31
Figura 2.18 Actuador neumático.....	32
Figura 2.19 Actuador eléctrico.....	33
Figura 2.20 Servomotores	34
Figura 2.21 Motores paso a paso.....	34
Figura 2.22 Motor de CC	35
Figura 2.23 Actuador lineal.....	35
Figura 2.24 Relevó.....	36
Figura 2.25 Selenoide.....	37
Figura 2.26 ESP32.....	38
Figura 2.27 Diagrama de bloque funcionamiento ESP32.....	39
Figura 2.28 Cloud computing.....	40
Figura 2.29 AWS (Amazon Web Services)	41
Figura 2.30 Amazon Timestream	44
Figura 2.31 DynamoDB	45
Figura 2.32 Componentes y funcionalidad de mediante la utilización de Real-Time IoT Device Monitoring con Kinesis Data Analytics.....	46
Figura 2.33 Grafana Cloud	49
Figura 2.34 Grafana Cloud	50
Figura 3.1 Prototipo IoT aplicado a la avicultura.....	54
Figura 3.2 Esquemático del prototipo IoT aplicada a la avicultura	54
Figura 3.3 ESP32 Wifi Core.....	56
Figura 3.4 Convertidor USB a Serial	57
Figura 3.5 Reguladores y protección.....	57
Figura 3.6 Módulo GSM para envío de SMS	58
Figura 3.7 Conectores para los sensores y actuadores (relés y controladores de caga de AC)	58
Figura 3.8 Esquemático del MIC-29302.....	59

Figura 3.9 Esquemático del LD1117ADT33TR	59
Figura 3.10 Esquemático del LM2576	60
Figura 3.11 Esquemático del ESP32	60
Figura 3.12 Esquemático del SIM800C	61
Figura 3.13 Esquemático del CP2102	61
Figura 3.14 Esquemático de las conexiones de los sensores	62
Figura 3.15 Esquemático del ADC ADS1115	62
Figura 3.16 Esquemático del Level Shifter	63
Figura 3.17 Esquemático del AC Controller	63
Figura 3.18 Diseño de placa electrónica	64
Figura 3.19 Diseño 3D placa electrónica	64
Figura 3.20 Frontal de placa electrónica	65
Figura 3.21 Diagrama de pistas de placa electrónica	66
Figura 4.1 Montaje de sensores de temperatura y humedad relativa	67
Figura 4.2 Prototipo IoT avicultura vista frontal	68
Figura 4.3 Prototipo vista lateral	68
Figura 4.4 Maqueta con viruta	69
Figura 4.5 Colocación de sensores	69
Figura 4.6 Colocación de sensores	70
Figura 4.7 Colocación de sensores	70
Figura 4.8 Colocación de sensores	71
Figura 4.9 Colocación de ventilador	71
Figura 4.10 Conexión de ventilador con el prototipo	72
Figura 4.11 Etiquetado de maqueta	72
Figura 4.12 Conexión del prototipo con los sensores	73
Figura 4.13 Conexión de los sensores y ventilador al prototipo IoT Avicultura	73
Figura 4.14 Interfaz de la consola de AWS IoT para los objetos	76
Figura 4.15 Interfaz de consola AWS para los certificados	76
Figura 4.16 Interfaz de consola AWS para las políticas para el prototipo	77
Figura 4.17 Regla de AWS IoT	78
Figura 4.18 Regla Data_Sensor	78
Figura 4.19 Acciones de la regla Data_Sensor	79
Figura 4.20 Configuración de la Acción de la regla Data_Sensor	79
Figura 4.21 Tabla sensores en DynamoDB	80
Figura 4.22 Información general de la tabla sensores	80
Figura 4.23 Métrica de CloudWatch de la tabla Sensores	81
Figura 4.24 Latencia en la tabla sensores	82
Figura 4.25 Exportaciones a S3	83
Figura 4.26 Elementos de la tabla sensores	84
Figura 4.27 Elementos de la tabla sensores	84
Figura 4.28 acciones de la tabla sensores	85
Figura 4.29 Flujo de DynamoDB	85
Figura 4.30 Información de CloudWatch de la regla "Data_sensor"	86
Figura 4.31 Usuario IAM	86
Figura 4.32 Configuración de Temperatura1 en Grafana	87
Figura 4.33 Configuración de Temperatura2 en Grafana	87
Figura 4.34 Configuración de Suelo1 en Grafana	88
Figura 4.35 Configuración de Suelo2 en Grafana	88
Figura 4.36 Configuración de Humedad1 en Grafana	88
Figura 4.37 Configuración de Humedad2 en Grafana	89
Figura 4.38 Configuración de HumedadSuelo1 en Grafana	89

Figura 4.39 Configuración de HumedadSuelo2 en Grafana.....	89
Figura 4.40 Configuración de Luz1 en Grafana	90
Figura 4.41 Configuración de Luz2 en Grafana	90
Figura 5.1 Ejecución del monitoreo de Sensores desde el IDE en Arduino.....	91
Figura 5.2 SMS de alerta recibidos	93
Figura 5.3 Monitoreo de los sensores desde AWS IoT	94
Figura 5.4 Acceso Grafana desde TimeStream.....	95
Figura 5.5 Registros de los datos mediante Grafana.....	96
Figura 5.6 Monitorización de Temperatura ambiente 1 y 2.	96
Figura 5.7 Monitorización de humedad relativa 1 y 2.	96
Figura 5.8 Monitorización de humedad de suelo 1 y 2.	97
Figura 5.9 Monitorización de temperatura de suelo 1 y 2.....	97
Figura 5.10 Monitorización de intensidad de luz 1 y 2.....	97
Figura 5.11 Alarmas de sensores.....	97

Índice de tablas

Tabla 2.1 Temperaturas de aire recomendadas de la fuente de calor durante la crianza de pollos de engorde.....	15
Tabla 2.2 Características técnicas del DTH11	27
Tabla 4.1 Temperatura ambiente y humedad los primeros 35 días del pollito en el galpón.....	75

Introducción

Las tecnologías del internet de las cosas aplicadas a la avicultura es un campo aún poco explorado en el Ecuador, debido a diferentes variables como el acceso limitado del internet en las zonas rurales del país, la falta de conocimiento técnico de los granjeros rurales criadores de aves, y el desarrollo de tecnologías aplicadas a la avicultura a nivel local por parte de la empresa privada, por estos motivos no hay un despliegue para la tecnificación de la crianza de pollos en el Ecuador como lo hacen en países desarrollados.

Una correcta tecnificación en la crianza de pollos de granja es indispensable para evitar la alta mortalidad que tienen estas aves por razones climáticas como altas o bajas temperaturas, excesiva humedad de ambiente y suelo y la falta de luminosidad cuando son pollos pequeños.

En este contexto se plantea el desarrollo de un prototipo IoT (Internet de las cosas) para la monitorización de parámetros ambientales aplicados a la crianza de pollos, utilizando hardware de bajo costo y servidores web en la nube. Con esta investigación se consigue fomentar el uso en los avicultores de la tecnología aplicada a la crianza de pollos.

Para el prototipo se utilizará elementos electrónicos como sensores, actuadores y una conexión de internet con la cual se logrará enviar datos como la temperatura del ambiente y la humedad ambiente del galpón de pollos, la cantidad de luz y la humedad del suelo para monitorizar a distancia mediante tecnologías en la nube utilizando el internet de las cosas. Con esta información recopilada se puede tomar decisiones para bajar o elevar la temperatura mediante actuadores que habilitarán ventiladores en el galpón de pollos, así mismo se controlará la cantidad de luz que hay en el galpón, se monitorizará si el piso está húmedo o en el ambiente hay mucha humedad que pueda afectar a los pollos de engorde.

Para realizar estas pruebas se implementará un prototipo de galpón con pollos y se monitorizará mediante los sensores el correcto crecimiento de los pollos. Como beneficio adicional este prototipo ayudará a los estudiantes de la carrera de ingeniería en telecomunicaciones poner en práctica sus conocimientos en el campo del internet de las cosas.

La clasificación de este trabajo de investigación es la siguiente: el primer capítulo explica el problema, antecedentes, importancia y alcance, demarcación, beneficiarios de la propuesta, soluciones propuestas y metas. El segundo capítulo es la base teórica del radar y el tercer capítulo es el marco metodológico y los tipos de investigación. El capítulo 4 trata de los resultados de la encuesta. El capítulo 5 trata del análisis de resultados, conclusiones y recomendaciones.

1. El problema

1.1. Descripción del problema

Las aves son un alimento indispensable en el consumidor ecuatoriano, es por esto por lo que muchas familias de las zonas rurales del Ecuador se dedican a la crianza de aves para el consumo interno o para la venta, sin embargo, esta crianza no es tecnificada y muchas aves mueren en su etapa de desarrollo.

La temperatura es un factor indispensable en la crianza de las aves, ya que el pollo debe criarse en un ambiente alrededor de los 28 a 32 grados dependiendo de su edad y tamaño y con una temperatura de suelo alrededor de los 28 grados centígrados, según estudios anteriores.

De igual manera la intensidad lumínica debe ser moderada por ejemplo para galpones con ambiente controlado es suficiente mantener entre 5 a 7 lux en recría y de 10 a 15 lux en producción.

Para el negocio de la crianza de aves las granjas avícolas utilizan galpones con ventanales que se pueden abrir y cerrar o utilizan ventiladores extractores de calor o ventiladores que ingresan aire fresco para aclimatar a las aves. El encendido y apagado de ventiladores o abrir y cerrar ventanales es un proceso manual que se hace siempre y cuando se supervise constantemente a las aves.

En este contexto, se presenta este trabajo de investigación que permita la tecnificación de la crianza de pollos de granja mediante herramientas del internet de las cosas, hardware de bajo costo y servidores web en la nube para el monitoreo de parámetros ambientales como temperatura ambiente, humedad ambiente, humedad del suelo e intensidad lumínica y que a través de datos obtenidos puedan activarse de manera automática un control de luz y de ventiladores para la regulación tanto de la luz como de la temperatura y humedad en la granja de pollos

1.2. Antecedentes

La alta tasa de mortalidad de los pollos de engorde criados en granjas en la región de Ecuador ha causado pérdidas económicas a las familias dedicadas a la cría y venta de pollos para el consumo local. Esta mortalidad puede estar relacionada a muchos factores como altas o bajas temperaturas en el criadero de pollos, falta de luminosidad o alta luminosidad, excesiva humedad de suelo y de ambiente.

De acuerdo con anteriores estudios la baja luminosidad en las granjas de pollos no permite su correcto desarrollo y posterior mortalidad. (Avicultura.info 2020)

Parámetros ambientales como altas o bajas temperaturas, humedad excesiva y falta de luminosidad afectan al desarrollo correcto del pollo de engorde. (Elsitioavicola.com 2012)

La falta de tecnificación en la crianza de pollos ha implicado que muchas familias pierdan una cantidad considerable de pollos que mueren por altas temperaturas o por la humedad de suelo.

En base a la investigación de otros proyectos y la falta de tecnificación en la crianza de pollos se plantea este proyecto de innovación de IoT aplicado a la crianza de aves, el cual monitorizará las medidas como la temperatura y saturación del ambiente, humedad de suelo, cantidad de luz del galpón de pollos para una mejor toma de decisiones durante la crianza de estas aves.

Con esta investigación la cual abarca la tecnificación de granjas avícolas utilizando hardware de bajo costo y tecnologías en la nube se pretende ayudar a las familias del campo de la costa ecuatoriana que se dedica a la formación de pollos de engorde para el consumo local. De igual manera los estudiantes de la carrera de ingeniería en telecomunicaciones se beneficiarán ya que pondrán en práctica sus conocimientos de telecomunicaciones al aplicarlos al IoT y a los servicios en la nube.

1.3. Importancia y alcance

El proyecto de investigación tiene una importancia muy alta para los granjeros rurales de la costa ecuatoriana que tienen como sustento familiar la crianza de pollos, si se aplica una correcta tecnificación en la crianza de estos pollos la economía de muchas familias rurales mejorará debido a que se evitará pérdidas que afecten su estabilidad. Al controlar parámetros ambientales durante el desarrollo de los pollos se logrará evitar un mal desarrollo del pollo inclusive su muerte.

El alcance del proyecto está aplicado al desarrollo de un prototipo IoT para el monitoreo de parámetros ambientales aplicados a la crianza de pollos de granja utilizando electrónica de bajo costo y servidores web de AWS (Amazon web services).

1.4. Delimitación

1.4.1. Delimitación temporal

El tiempo estimado para el diseño, implementación y pruebas de funcionamiento del prototipo es de 6 meses, se tiene previsto en finalizar en el mes de marzo del 2021.

1.4.2. Delimitación espacial

Debido a la emergencia sanitaria por motivo de la pandemia del COVID-19 y siguiendo las indicaciones de la dirección de carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana, sede Guayaquil, la implementación del sistema IoT de monitorización ambiental para la crianza de pollos será en el domicilio del egresado Félix Mora autor del proyecto de tesis, en la ciudad de Durán, mediante una maqueta a escala que servirá para poner a prueba el prototipo.

El testeo y análisis de resultados se lo hará en tiempo real de acuerdo con el diseño del prototipo IoT toda esta información obtenida de los sensores del prototipo serán administrados vía web en servidores AWS (Amazon Web Services) y controlados mediante acceso remoto a través del internet.

1.4.3. Delimitación académica

Este trabajo de investigación donde el resultado es un prototipo IoT funcional y operativo será uno de los instrumentos esenciales para el proceso de las capacidades y destrezas de los futuros ingenieros en telecomunicaciones de la Universidad Politécnica Salesiana sede Guayaquil. Con este trabajo se pone en práctica los conocimientos adquiridos en materias como redes de computadora, electrónica digital, comunicaciones digitales y medios de transmisión.

1.5. Explicación del problema

En las probables razones para la mortalidad de los pollos de engorde a lo largo de su crianza en los galpones es la carencia de control de las fronteras del medio ambiente a lo largo de su desarrollo a partir de que son pollitos hasta una vez que ya permanecen listos para el consumo humano.

De acuerdo con estos causales se plantea la relación de variables que se medirán en este proyecto de investigación, como la temperatura ambiente, humedad relativa ambiente, humedad de suelo, intensidad lumínica.

Por medio de la inspección de las fronteras del medio ambiente de un galpón de crianza de pollos de engorde se tecnificará el desarrollo de los pollos para consumo. Esto se logrará con dispositivos como sensores y actuadores de bajo costo como el ESP32, sensores ambientales y actuadores que activarán ventiladores para el control de la temperatura y humedad, así como triggers para el control lumínico en ausencia de luz solar. Estos parámetros serán monitorizados mediante la web a través de servicios en la nube como el ofrecido por AWS IoT.

1.6. Objetivos

1.6.1. Objetivo general

Diseñar e implementar un prototipo IoT para el monitoreo de parámetros ambientales aplicados a la avicultura para la crianza de pollos de granja utilizando ESP32 y servidores web en la nube con AWS (Amazon Web Services).

1.6.2. Objetivos específicos

- Diseñar red de sensores de temperatura y humedad ambiental, humedad de suelo e intensidad lumínica para dispositivo IoT de monitorización de parámetros ambientales para la crianza de pollos de granja.
- Configurar servicios en la nube para el almacenamiento y monitorización de la información de los sensores con Amazon Web Services.
- Diseñar un sistema de alerta SMS para la monitorización de umbrales de los parámetros de temperatura, humedad, humedad de suelo y fuerza de luz para la crianza de pollos de granja.
- Realizar manual de práctica del prototipo IoT de monitorización ambiental para la crianza de pollos de granja.

2. Fundamentos teóricos

2.1. Avicultura

Es una forma de criar aves (pato, pavo, ganso, parte, faisán, codorniz, avestruz, etc.). Suelen satisfacer mercados establecidos, como la venta de huevos y carne. También hay especies exóticas que no han sido domesticadas. El propósito de la producción avícola es encontrar especies menos valiosas y utilizar el patio o el área para criaturas rurales y suburbanas autosuficientes (ecured.cu 2021).

En Ecuador, hay 1.819 ranchos avícolas. La industria avícola forja cerca de 32.000 principios directas de trabajo, 220.000 indirectas, alrededor de US \$ 2 mil millones al año, lo que representa el 16% del PIB agropecuario.2% del PIB (DAVE. Diana Espín, Director Ejecutivo de la Compañía Nacional de Avicultores del Ecuador, CONAVE). Existe más de la mitad de la fabricación de pollo del país se agrupa en las provincias de Guayas (22%), Pichincha (16%) y Santo Domingo de los Tsáchilas (14%). Según un artículo de la revista Avinews (octubre de 2017), Ecuador produce aproximadamente de 230 a 250 millones de pollos de engorde cada año. El gasto per cápita fue de 30 a 32 kg en 2017 y se triplicará en 20 años. Este nivel de gasto es superior al del cerdo: 12 kg, vacuno y pescado: 10 kg (Planv.com.e 2021).



Figura 2.1 Granja de pollos
Fuente: (Planv.com.e 2021)

2.2. Características de los pollos de engorde

Es el pollo en su etapa inicial de vida, el cual es criado en granja y engordado. Su ingesta de alimentos consta de vitaminas y proteínas que se le da o se llama balanceado, para que finalmente del proceso tenga el más grande peso y al final se sacrifica. Su fin es conseguir el crecimiento de la producción de la granja avícola y por lo tanto el crecimiento de mercado.

2.2.1. Alimentación del pollo de engorde

La ingesta de alimentos es la etapa de mayor relevancia dentro del proceso del pollo, debido a que constituye mínimo el 70% del precio de producción y por ende es el componente esencial para tener en cuenta. Una ingesta de alimentos correcta asegurará en el pollo una buena constitución del cuerpo referente a músculos, huesos y grasas.

Cada programa de alimentación es relativo y varía dependiendo de las necesidades que se tienen para obtener del pollo (peso del pollo, porcentaje despresado, asaderos, subproductos para carnes frías, etc.). De manera práctica se suministra 1500 gramos de alimento iniciación al macho y 1200 gramos a la hembra, con el fin de desarrollar estructuralmente mejor al macho para que alcance todo su potencial genético. El alimento se suministra en presentación en harinas o en presentación crombelizado para la fase de iniciación, aunque esto depende también del clima, altura y formulación. Solo se suministra en presentación de pellets en la última semana (EcuRed 2021).

Se controla con la suspensión de este para que logre asimilar todo lo cual ha consumido, es por esto el valor de la iluminación nocturna. En épocas de calor, lo que es una fuente de estrés y tiene como resultado instantáneo un bajón importante en la ingestión de alimentos y afecta en el aumento de los pollos, es primordial el control de la ingesta de alimentos, continuamente que se logre, tomando medidas de reservas preventivas a probables atrasos (por ejemplo, de encargo) u otras necesidades. Comúnmente para un estudio de balanceado se toma presente las proteínas la cual fluctúa entre 18% y 24%, la grasa en 3% y 4%, la fibra entre 4% y 5%, y las calorías, componentes bastante relevantes para balancear una dieta de pollo. Grosso modo, la materia prima elemental incluye maíz y sorgo, que representan 60%, harina de soya, harina de pescado, polvillo, aceite de palma, sal y compuesto de vitaminas y minerales.



Figura 2.2 Pollo de engorde
Fuente: (Hablemosdeaves.com 2021)

2.2.2. Anatomía del pollo de engorde

Los Pollo de Engorde de usos comerciales modernos, ejemplificando, los cruces de Cornualles y los Cornish – Rocks, son los más seleccionados artificialmente y criados para la producción de la carne de forma eficiente a una enorme escala. Se resaltan por tener las tasas de aumento bastante rápidas, una alta tasa de conversión alimenticia y además de bajos niveles de actividad. Los Pollo de Engorde comerciales modernos se crían para conseguir un peso aproximado de sacrificio de cerca de unos 2 kilogramo en cuestión de 35 a 49 días.

Como resultado, la conducta y la fisiología de los pollos de engorde que son criados para la carne son los de las aves inmaduras, en vez de los adultos. Se han desarrollado las cepas orgánicas de aumento lento y el aumento independiente que alcanzan el peso de sacrificio en medio de las 12 y 16 semanas de edad.

Los pollos de engorde tradicionales tienen las plumas de color blancas y su piel es amarillenta. Actuales los estudios de los genes han demostrado que el gen de la dermis amarilla se alcanzó a integrar a las aves domésticas mediante la hibridación con el junglefowl gris (*G. sonneratii*). Los cruces modernos además son favorables para la producción de la carne ya que carecen del típico "pelo" que poseen muchas de las razas que requiere chamuscarse desde el desplume. Los pollos de engorde, tanto los machos como las hembras, son criados para su carne. Un pollo suele medir unos 40 cm de altura en promedio, aunque es un ave, no tiene mucha suerte en el vuelo.

El macho generalmente se llama gallo, la hembra se llama gallina y su pollito amarillo

esponjoso se llama pollito. En la naturaleza, pueden sobrevivir durante 4-5 años, pero en una granja familiar automatizada, no pueden durar varios años. El pollo registrado con la vida útil más larga sobrevivió durante unos 16 años.

Son una especie de pájaro sociable y son más felices cuando están rodeados de sus compañeros. En un grupo de gallinas, el número de gallinas puede ser ilimitado, pero generalmente solo un gallo es el macho dominante, cuando piensa que es una amenaza para él, amenaza con expulsar a otros gallos del grupo. El gallo dominante tiene derecho a aparearse a las gallinas que tiene a su cuidado. (Hablemosdeaves.com 2021)



Figura 2.3 Cría del Pollo de Engorde
Fuente: (Agrotendencia.Tv, 2021)

2.2.3. Características de una buena calidad de pollito

- Apropiadamente seco y de plumón prolongado.
- Fanales grandes, brillantes y activos.
- Pollitos activos y alertas.
- Ombligo completamente cerrado.
- Las patas deben ser brillantes a la vista y cerosas al contacto.
- Las articulaciones tibiotarsianas no deben estar inflamadas.
- Los pollitos deben estar libre de deformaciones (patas torcidas, cuellos doblados o pico cruzados). (EcuRed 2021)

2.3. Sectores en el Ecuador adecuados para la avicultura

En todo el país, el 84,7% de la avicultura se produce en granjas avícolas y 15,3% en el campo. Sin embargo, algunos tipos son gallos, gallinas, patos y especies de patos especialmente criados en el campo. Solo criados como pollos en granjas avícolas Criador, codorniz y avestruz. Las aves más reproducidas en el país son polluelos, gallinas jóvenes, pollitos y 71% de pollos y 16% de gallinas ponedoras, como se aprecia en la siguiente figura. La cría de avestruces en Ecuador no es importante, 5.078 unidades de cría se registraron en 2014, solo en 2019 A 90.000 unidades sin criar en años anteriores (Cedia 2020).

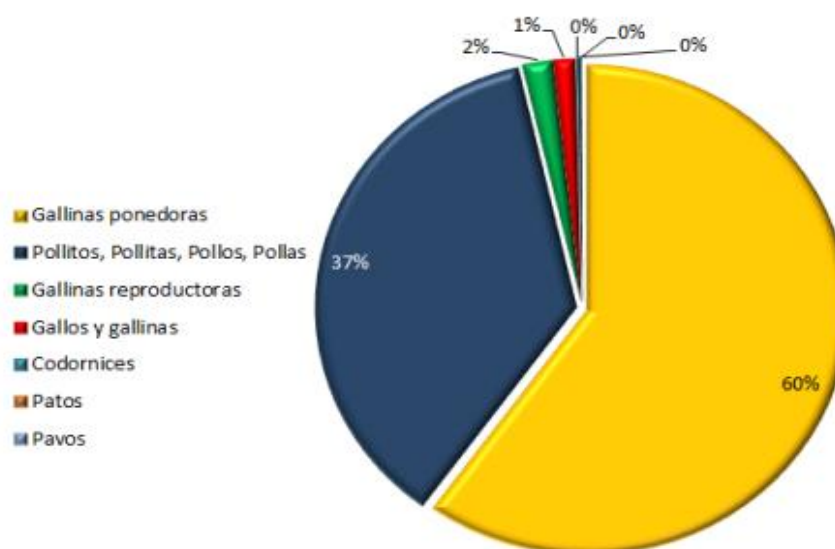


Figura 2.4 Sectores en el Ecuador adecuados para la avicultura

Fuente: (Sector Avícola Ecuador, 2019)

La industria avícola se desenvuelve en 24 provincias del país. El 80% de la avicultura del país se agrupa en 9 provincias, principalmente Guayas, Pichincha, Tungurahua (Tungurahua), Santo Domingo (Santo Domingo de los Tsáchilas), Manabí, El Oro, Cotopaxi, Imbabura y Pastaza. (Sector Avícola Ecuador, 2019)

Tungurahua, convirtiéndose una de las provincias más particulares, se conoce que el 60% de las aves criadas en el huerto y granjas avícolas se enfoca en las gallinas ponedoras. Las gallinas reproductoras cesaron la elaboración a partir del año 2015, sin embargo, la cría de codornices aumentó más del 100% entre el 2014 y 2019. No se registra la producción de avestruces en la provincia y la cría de pavos es muy mínima con 170 unidades del promedio anual. (Sector Avícola Ecuador, 2019)

El 97% de la industria avícola del país tiene la intención de vender el 3% restante para uso propio. La especie más cara es el gallo, pollo y pato. Así mismo, en Tungurahua, el comisión destinada a las ventas es superior al autoconsumo, y las mayores en ventas dan en pollitos, pollitas, gallinas y gallos. El mayor autoconsumo de Tungurahua se concentra en gallos y gallinas. Según datos que la Compañía Nacional de Avicultores de Ecuador-CONAVE, el consumo anual de pollo per cápita es de 30,43 Kg en el 2019 con un desarrollo del 16% en el 2018, donde el consumo per cápita fue de 26,3 Kg por año (Sector Avícola Ecuador, 2019).

Hoy, con el desarrollo de nuevas tecnologías y soluciones, la industria avícola del país continúa desarrollándose en la dirección de la producción intensiva y técnica. Los pilares de la producción sostenible se basan básicamente en la genética, la nutrición, el manejo y la salud animal, por eso es muy importante contar con medicamentos veterinarios de primera línea.

Hoy en día, los productores de aves de corral pueden confiar en el asesoramiento adecuado para establecer una granja productiva (pollos de engorde o ponedoras). Este trabajo comienza con la selección de la ubicación de la casa y se extiende hasta la construcción del galpón.

Antes de iniciar el proyecto avícola, muchas personas enfatizaron la necesidad de entender el mercado, porque este será la base de cualquier fuente de ingresos que haga viable el negocio.

El mercado nacional está bastante desarrollado y tiene un alto contenido técnico, pero actualmente enfrenta el problema del abastecimiento de pollos y huevos fértiles, Brasil abastece una gran cantidad de estos pollos y huevos fértiles. En otras palabras, es difícil producir pollitos, lo que significa que el fabricante produce menos pollitos por lote.

A pesar de esta situación, las perspectivas son alentadoras ya que las expectativas de crecimiento de la industria son buenas. Si bien seguimos siendo la región con el menor consumo anual de carne de cerdo y aves per cápita, es probable que lleguemos a otros países de la región debido a nuestras condiciones climáticas favorables para la producción de granos, para la cría y producción avícola. (ABC RURAL, 2013)



Figura 2.5 Tecnificación en la Avicultura

Fuente:(ABC RURAL, 2013)

El fragmento avícola en el Ecuador, es una condecoración que ha crecido pausadamente, solamente entre el 2018 y 2019, la cifra de aves criadas en campo y planteles avícolas creció 27%. El consumo de carne de pichón es esencial en la abstinencia de los ecuatorianos y forma parte de la barquilla sencillo básica. De entente con la compañía de las Naciones Unidas para el alimento y el quehacer (FAO), la adquisición de carne de aves ocupa la segunda sede a grado ecuménico luego de la carne de cerdos.

El costo de la libra de pollo en el distrito es de entre \$1,20 y \$1,25. De completado la producción de aves en el Ecuador no resguarda la petición interna, por lo que es necesaria la transacción de machos y hembras reproductoras pesadas de engorde, ponedoras comerciales y machos y hembras reproductores livianos. (Cedia 2020)

Tipo de ave	2014	2015	2016	2017	2018	2019	Total general
Pollitos, Pollitas, Pollos, Pollas	61.170.786	33.285.581	34.966.344	29.178.940	30.352.273	38.813.881	227.767.805
Gallinas ponedoras	8.916.554	9.999.484	9.344.355	8.355.801	7.589.624	8.936.553	53.142.372
Gallos y gallinas	5.305.102	2.639.893	3.479.422	3.346.539	3.197.199	4.775.794	22.743.950
Gallinas reproductoras	1.841.048	2.520.423	1.603.925	1.822.851	2.269.552	2.399.881	12.457.681
Patos	655.223	450.749	524.959	484.214	492.494	585.087	3.192.725
Pavos	520.270	291.631	280.846	212.115	221.003	228.959	1.754.824
Codornices	484.461	246.326	115.717	100.813	61.227	83.375	1.091.918
Avestruces	5.078	300	250	-	-	90.000	95.628
Total general	78.898.522	49.434.388	50.315.817	43.501.272	44.183.372	55.913.531	322.246.903



Figura 2.6 Tipos de aves criadas en el Ecuador
Fuente: (Cedia 2020)

2.4. Factores que influyen en la crianza de pollitos de granja

En la formación de pollitos es crítico conservar la temperatura adecuada, en especial a lo largo de sus 2 principales semanas de vida. Al nacer, el pollito está mal preparado para regular sus procesos metabólicos y mantener el control de correctamente la temperatura de su cuerpo humano.

Como consecuencia, el pollito recién nacido es dependiente de la temperatura ambiental para conservar la temperatura del cuerpo óptima. Si la temperatura reduce, además lo va a hacer la temperatura del cuerpo del pollito. Asimismo, si se incrementa la temperatura medioambiental, además incrementará la temperatura del cuerpo del pollo.

Demasiado frío o calor a lo largo de este lapso determinante puede ser en un pobre aumento, una mala conversión alimenticia y más grande susceptibilidad a patologías. Las prácticas idóneas de crianza tienen que conservar la temperatura del cuerpo del pollo para que no tenga que usar energía, para perder calor por medio del jadeo o para crear calor por medio de su metabolismo.

Las investigaciones han demostrado que el pollito desarrolla la capacidad de regular su temperatura corporal alrededor de los 12 y 14 días de edad.

El pollo se puede estresar de forma fácil si su temperatura del cuerpo reduce o se incrementa tan solo un nivel. Cuando cambia su temperatura del cuerpo, el ave tratará de compensarla y en varios casos esto quiere decir que va a tener un impacto negativo en el rendimiento.

La temperatura corporal de un pollito en su primer día de edad es de aproximadamente 103°F (39°C), pero para cuando transcurren los primeros cinco días de edad la temperatura corporal es 106°F (41°C), igual que el adulto.

Las temperaturas extremas (altas o bajas) a menudo provocan la mortalidad de los pollitos, pero incluso un leve enfriamiento o sobrecalentamiento puede afectar el rendimiento de los pollitos jóvenes sin causarles la muerte.

Mientras que los pollitos toleran las temperaturas altas mejor que las aves adultas, las temperaturas altas durante largos periodos de vigencia incrementan la mortalidad y tienen una patada negativa en el beneficio.

Las investigaciones han demostrado que los pollitos sometidos a una temperatura fría tienen dificultades con sus sistemas inmunológico y eupéptico. Como resultado, pollitos estresados por el sosegado crecen a excepción de y tienen mayor susceptibilidad a las enfermedades. Los pollitos estresados por el pausado culparán una mayor incidencia de ascitis, un trastorno metabólico que se traduce en benjamín lucro, mayor mortalidad y mayores decomisos en la apostura de procesamiento.

En aprendizaje de investigación, donde se criaron asociados de pollitos a 80°F o 90°F (27°C o 32°C), los pollitos criados bajuno temperaturas más cálidas tuvieron mejores ingresos de balancín, mejor conversión alimenticia y mejores condiciones de ánimo. Los pollitos criados abajo de 80°F (27°C) experimentaron una última transformación que la receta de lactancia con una temperatura más inscripción.

Los pollitos criados bajo esas temperaturas no alcanzaron el peso vivo y resultaron pesando indemne en la edad de comercializarlos que las aves que se criaron correctamente. No solo que los pollos expuestos a bajas temperaturas de amamantamiento tienen menores tasas de transformación, hado que asimismo gastarán más alimento para mantenerse calientes, reduciendo la eficacia de los alimentos y agigantando los costos de sustento. (Elsitioavicola.com 2012)



Figura 2.7 Pollitos en galpón
Fuente: (Cedia 2020)

2.4.1. Temperatura y rendimiento del pollo

Uno de las metas a lo largo de la crianza es conservar a los pollitos en su región de bienestar, o sea, que no usen energía para triunfar o perder calor, para conservar la temperatura del cuerpo.

Una vez que las aves se mantienen en temperaturas del medio ambiente por arriba o por abajo de su región de bienestar, tienen que gastar más energía para conservar la temperatura del cuerpo. Esta energía extra se suministrará en última instancia por medio de los alimentos consumidos. Por consiguiente, se usará la energía del alimento para conservar la temperatura del cuerpo, en vez de emplearla para el incremento y desarrollo, lo que resultará en una mala conversión alimenticia. De esta forma es como la temperatura del medioambiente juega un papel fundamental para establecer el precio de crear un kilogramo de carne o para criar una pollona reproductora.

La crianza adecuada no sólo consiste en mantener la temperatura debida, sino también en emplear las prácticas de buen manejo. Las temperaturas empleadas en la crianza variarán dependiendo de si la fuente de calor es un calentador de aire, una criadora infrarroja convencional o una criadora radiante (Tabla 1). Tenga en cuenta que las temperaturas en la tabla 1 se refieren a temperaturas del aire diseñadas para proporcionar una temperatura de piso de 90°F (32°C).

Al criar pollitos la temperatura de la cama es crucial. Los estudios sugieren que la temperatura promedio del piso debe ser 90°F (32°C) el día se colocan los pollitos en el galpón.

Los calentadores de aire forzado requieren ajustes de temperatura más altos, ya que ellos calientan el aire que calienta el piso. Una criadora infrarroja convencional de campana dirige aproximadamente 40 por ciento de su calor al piso y 60 por ciento al aire. Las criadoras infrarrojas radiantes proyectan aproximadamente 90 por ciento de su calor al piso y 10 por ciento al aire. Debido a que las criadoras infrarrojas de campana y las criadoras radiantes dirigen más calor al piso, la temperatura del aire necesaria para lograr la temperatura deseada en la cama es menor que la requerida para los calentadores de aire forzado. (Elsitioavicola.com 2012)

Tabla 2.1 Temperaturas de aire recomendadas de la fuente de calor durante la crianza de pollos de engorde

Temperatura del Aire			
Día	Calentador de Aire Forzado²	Criadora Infrarroja Convencional³	Criadora Infrarroja Radiante⁴
0	93°F (34°C)	90°F (32°C)	88°F (31°C)
3	90°F (32°C)	88°F (31°C)	86°F (30°C)
7	87°F (31°C)	86°F (30°C)	84°F (29°C)
14	83°F (28°C)	85°F (29°C)	82°F (28°C)
21	78°F (26°C)	80°F (27°C)	77°F (25°C)

¹Temperaturas basadas en las empleadas actualmente por empresas de pollos de engorde
²Medido a la altura del pollito.
³Medido a la altura del pollito, a 30 cm del borde de la criadora.
⁴Medido a la altura del pollito, a 1.20 m del borde de la criadora.

Fuente: (Elsitioavicola.com 2012)

Los galpones de pollos permanecen especialmente diseñados para aprobar el intolerante cuidado de la atmósfera que está de manera directa en torno al pollito. En operaciones de crianza de pollos comerciales, los galpones de los pollos permanecen aislados y equipados con sistemas de ventilación mecánica para reservar las temperaturas del galpón en 5 grados F de la temperatura deseada, independientemente de la temperatura fuera. Las temperaturas óptimas para la lactancia de pollitos se han desarrollado por atmósfera de varios etapa de indagación y experiencia en el recorrido.

Una vez que se recibe la temperatura correcta de la caseta y los pollitos permanecen bien administrados, ellos deben estar distribuidos a lo largo del galpón y no arrimados unos con otros, o sentados en las bandejas de los comederos.

Cuando se obtiene la temperatura adecuada de la garita y los pollitos están bien administrados, ellos deben radicar distribuidos en todo el galpón y no acurrucados unos con otros, o sentados mayormente en las bandejas de los comederos.

El mejor método para supervisar la felicidad del pollito es notar su proceder y habitual la temperatura consecuentemente. Al sentir un galpón de pollos de aumento, los pollitos deben vivir distribuidos uniformemente en todo el galpón.

Cuando los pollitos tienen machucho se les ve acurrucados (además pegados al burladero adyacente) y tienden a sentarse en las bandejas de los comederos. Los pollitos que tienen pasión se alejan de las criadoras infrarrojas y calentadores, jadean y se separan de la parvada tratando de acatarrarse.

Muchos galpones tienen controles ambientales que observan la temperatura del galpón y prenden y apagan las criadoras infrarrojas y los calentadores según sea necesario. Los controles regulan también el tiempo que opera una fuente de calor. Esta información puede utilizarse para localizar áreas del medioambiente del pollito que pueden tener excesivas pérdidas de aire o corrientes de aire.

Por ejemplo, si las criadoras infrarrojas o los calentadores cerca de la cortina del túnel funcionan más tiempo que otros en el galpón, puede indicar fugas de aire alrededor de la cortina o de la puerta al extremo de la pared. (El sitio avicola.com 2012)



Figura 2.8 Galpón de pollos
Fuente: (Cedia 2020)

2.4.2. Humedad relativa

La capacidad del aire para mantener la humedad depende de su temperatura. El aire caliente puede contener más humedad que el aire frío. El término humedad relativa se refiere al porcentaje de saturación de agua en el aire a una temperatura determinada. Los niveles de humedad afectan la capacidad de los pollos para enfriarse jadeando y afectan la producción de amoníaco.

Se recomienda mantener la humedad relativa entre el 50% y el 70% durante la temporada de crecimiento (incluido el período de alimentación). El amoníaco se produce debido a la descomposición de las heces en la cama por microorganismos. Las condiciones de polvo en el gallinero están relacionadas con una humedad relativa inferior al 50%.

Una humedad relativa del 70% o más proporciona las condiciones ambientales adecuadas para el crecimiento de microorganismos en la cama. A medida que aumenta el número de microorganismos, se producirá más amoníaco a partir de la fuente de nitrógeno en el material de guano.

El amoníaco es un gas que tiene un golpe negativo en el beneficio y la salud de los pollos. Los estudios han demostrado que el aumento de amoníaco puede dañar el sistema inmunológico y aumentar las enfermedades respiratorias en las aves. Los altos niveles de amoníaco durante la cría ralentizarán la tasa de crecimiento, pero no se pueden recuperar durante el resto del proceso de crecimiento. La producción de amoníaco se puede reducir controlando la humedad relativa y la humedad relativa se puede ajustar mediante ventilación. Se recomienda gestionar la tasa de ventilación del gallinero para mantener la humedad relativa entre 50% y 70% para minimizar la producción de amoníaco y polvo.

La exposición a agrupaciones de amoníaco tan bajas como 25 ppm en los primeros 28 días puede tener un impacto negativo en el peso a las 4 y 8 semanas de edad.

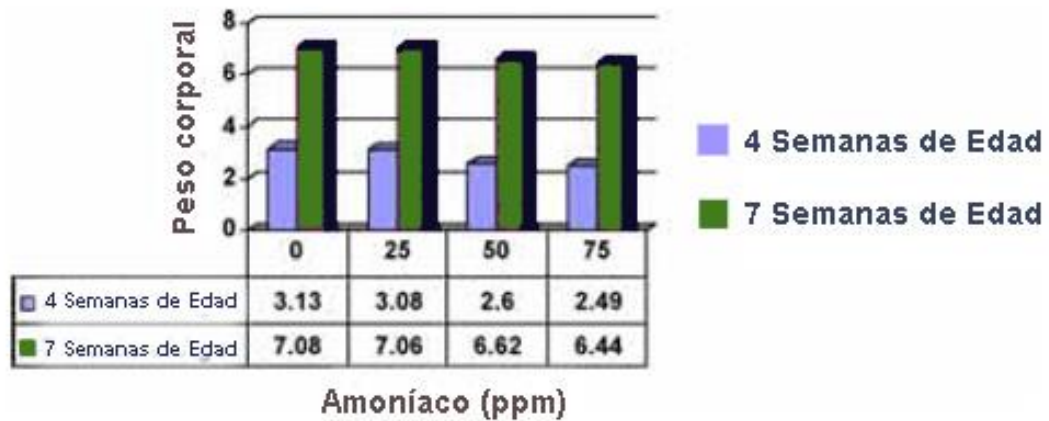


Figura 2.9 Niveles de amoniaco
Fuente: (Elsitioavicola.com 2012)

2.4.3. Ventilación

La ventilación es primordial para ajustar la temperatura y eliminar el dióxido de carbono, el amoniaco, otros gases, la humedad, el polvo y los olores.

El aire fresco debe introducirse de manera uniforme, completamente mezclado con el aire del gallinero y debe distribuirse adecuadamente por todo el gallinero. El patrón de flujo en el edificio es muy importante. La circulación de aire en el gallinero se realiza mediante la presión negativa en el túnel del gallinero.

El ventilador expulsa el aire de la casa, creando una presión negativa. El aire ingresa a través de una entrada de aire ubicada en la pared o el techo, y se guía a través del techo para mezclar el aire.

La mezcla del aire del exterior con el aire del interior evita que el aire frío se asiente cerca de la parvada de pollos y enfría la parvada de pollos.

Las entradas de aire y la velocidad del aire que ingresa a través de éstas son importante, para asegurar que el aire se mueva a lo largo del techo. Si las entradas se abren exageradamente, o si la velocidad del aire que completa es demasiado baja, el aire frío caerá al piso más rápido. Esto no solo producirá el dificultad para que las aves se enfriarán, sino que puede crear puntos fríos en paredes y piso.

A lo que el aire tibio entra en contacto con estos puntos fríos se formará condensación y se formarán puntos húmedos. Estos puntos húmedos pueden llevar a aumentar el endurecimiento de la cama, así como a una mayor producción de amoniaco.

Se deben explotar ventiladores de circulación para romper la estratificación de la

temperatura, y favorecer una temperatura más uniforme en toda la kiosco de aves. El tamizar el ademán más cariñoso al nivel de las aves no solo báculo a mantener las temperaturas del batallón y del asfalto de las aves, fortuna que además báculo a matar la humedad de la hamaca.

Se controla el funcionamiento de los ventiladores por ámbito de la temperatura, para proveer la temperatura deseada. Algunos ventiladores trabajan con temporizador para frecuente la humedad relativa y amamantar la buena calidad del talante cuando el galpón está a la temperatura deseada. El controlador ambiental de la caseta opera los ventiladores en colchoneta a ajustes en la temperatura y en el temporizador. Los controladores estrenarán entradas de comportamiento situadas en las paredes laterales o en el divisoria, para garantizar la entrada consecuente de ademán en la caseta. La cantidad que se abren las entradas está determinada por bloqueo estática.(Elsitioavicola.com 2012)

2.4.4. Iluminación durante la crianza

La luz es un componente fundamental durante la crianza, que no se debe minimizar. Los movimiento de los pollitos es mayor en energía de luz brillante que en baja intensidad de luz.

Durante la lactancia la luz debe poseer una fuerza de lo más nacarado para respaldar la movimiento de pollito y así ayudarles a demarcar alimento y líquido. Una vez que aprenden donde se encuentran el sostenimiento y el consumición (entre desganado y 10 días), la intensidad de la luz y la longevidad puede y debe reducirse. Los sistemas de luz deben diseñarse para barbechar un carencia de 25 lux (velas de 2.5 pies) o más al nivel de las aves. Muchas de las casetas de pollos de aumento que se construyen hoy son capaces de favorecer hasta 40 lux (velas de 4 pies) a nivel de los pollitos en el área de camada.

El diseño del sistema de luminosidad deberá licenciar arreglar la afectación y la persistencia de la luz según crecen las aves, y entregar una vigor de luz firme a grado de las aves. Normalmente las entendimiento funcionan 23 horas al día durante la crianza y la empaque de la luz es adagio.

La intención principal de la iluminación durante la crianza es garantizar que los pollitos estén activos y que busquen las fuentes de alimento y agua. (Elsitioavicola.com 2012)



Figura 2.10 Iluminación en galpones de pollos
Fuente: (Elsitioavicola.com 2012)

2.4.5. Conservación de energía

Asegúrese de que el control del sensor / termostato esté en la posición correcta: colocar el sensor / termostato demasiado cerca de la incubadora infrarroja o de los pollitos, o demasiado alto en relación con el piso, puede hacer que las aves se enfríen. Si los coloca demasiado cerca de las paredes laterales, de las cortinas o de los lugares donde entra el aire en la nave, puede producirse un consumo excesivo de combustible.

Use ventiladores de circulación o de aspas: use estos ventiladores para eliminar el aire caliente del techo, eliminar la estratificación de la temperatura y reducir el calor al nivel de los pollitos que necesitan, y reducir los costos de calefacción. Los estudios han demostrado que no importa qué tipo de sistema de calefacción se utilice, los ventiladores de circulación son eficaces.

Filo de actitud en el ático – embuchar disposición del ático en días frescos cuando sale el sol, puede quedar en mayores tasas de aeración sin mayores costos de gasolina. Esto permite a los productores beneficiarse el aire cálido que se encuentra en el ático, el cual becará a lactar la humedad relativa entre 50 y 70 por ciento. Para que las entradas de ademán del ático funcionen correctamente, la garita debe ser lo suficientemente hermética para desnudar por lo aparte 0.13 pulgadas de vinculación de brebaje en una experimentación de boicot estática. Si las casetas no pueden

desenfundar esta boicoteo estática, se debe efectuar el interés de hermetizar los galpones para disminuir la cercanías antaño de instalar entradas en el ático.

Borrar del mapa filtraciones: estampillar las fugas proporciona pelotón de dónde y cuánto aspecto llegará a la kiosco. Esto garantizará que el talante ingrese a través de entradas previstas y minimizará la decisión que las aves sientan sosegado, que la cama se endurezca y que la temperatura se estratifique durante el flato sosegado.

Use temporizadores de cinco minutos en sede de temporizadores de diez minutos: Esto reducirá la posibilidad que la temperatura del galpón baje, y como resultado los calentadores funcionarán indemne asamblea, subvencionará también a común los niveles de amoníaco y de humedad.

Limpie y repare las criadoras infrarrojas de manera habitual – El mantenimiento adecuado de la criadora infrarroja reducirá el monóxido de grafito y ésta carbonizará el carburante de manera más efectivo. Se debe soplar el polvo de las criadoras infrarrojas entre cada parvada. Mantenga limpios los orificios del quemador. Utilice para soldar el cuerpo debido de agujas para eludir enojar el bombeo del agujero y consumir combustible (Elsitioavicola.com 2012)

2.5. Avicultura y el Internet de las cosas

La aplicación de Internet de las cosas (IOT) logra transformar una rancho tradicional en una granja avícola perspica moderna y poseerá una gran marca en el crecimiento de los pollos. Este es el resultado de la indagación realizada en el departamento de ECE del Instituto de Conocimiento y Tecnología de Salem, India.

Esta investigación se centra en las tecnologías modernas utilizadas en las granjas avícolas para controlar todos los parámetros climático que inquietan el desarrollo de los pollos, como la temperatura, la humedad y el amoníaco.

En el desarrollo de los pollos, el clima brinda un papel vital. Las granjas avícolas inteligentes están diseñadas de manera que pueden cambiar el clima a través de ventilación, enfriamiento y extractores.

Los parámetros de temperatura, humedad, amoníaco y nivel de agua son monitoreados por un microcontrolador PIC. El valor del sensor monitoreado se carga

en la página web, y luego el responsable puede utilizar Internet para comprender el ambiente doméstico de la granja avícola a través de una máquina móvil o personal. (avicultura.info 2021)



Figura 2.11 IoT aplicada a la avicultura
Fuente: (Engormix.com 2021)

2.5.1. Granja avícola inteligente: solución integrada que utiliza una red basada en WSN y GPRS

Rupali B. Mahale y otros discutieron el tema del monitoreo de granjas avícolas mediante el uso de sensores inalámbricos (WSN) y redes basadas en GPRS. El sistema monitorea el agua, el nivel de alimento y las medidas ambientales incluida la temperatura y la humedad. (Avicultura.info 2018)

2.5.2. Granja inteligente con red de sensores inalámbricos

Vaibhavraj S. Roham y otros describieron redes de sensores inalámbricos para diseñar entornos inteligentes para monitorear y controlar varias medidas climáticas. (Avicultura.info 2018)

2.5.3. Granja avícola inteligente basada en Internet

K. Sravanth Goud y otros desarrollaron redes de sensores inalámbricos avanzados y tecnología de redes móviles, que pueden monitorear y examinar mecánicamente las medidas ambientales de las aves de corral. Esta individuo consigue monitorear las medidas ambientales enviando SMS al sistema. (Avicultura.info 2018)

2.5.4. Incorporar tecnología de detección inteligente en la industria avícola

Gerard Corkery y otros batallaron el uso de tecnología de sensores inteligentes en la producción avícola para monitorear medidas ambientales clave relacionados con la producción avícola, que contienen, entre otros, la temperatura del aire, la humedad relativa, la luz, la velocidad del aire y la característica del aire (especialmente la concentración de CO₂ y NH₃). (Avicultura.info 2018)

2.5.5. Diseño e implementación de relé inteligente basado en monitoreo remoto de amoníaco en granjas avícolas

Jawad K. Othman y otros utilizan relés inteligentes (relés inteligentes) en granjas avícolas que, en comparación con los dispositivos tradicionales, brindan mucha flexibilidad en el diseño y ejecución de sistemas de monitoreo y control, y reducen costos. (Avicultura.info 2018)

2.5.6. Uso del Raspberry Pi y Arduino para el monitoreo inteligente de la granja

Siwakorn Jindarat y otros diseñaron un sistema inteligente en sistemas integrados y teléfonos inteligentes para el manejo de aves de corral. Para solucionar esta dificultad, el autor utilizó Raspberry Pi y Arduino Uno. El procedimiento debe controlar los parámetros circundantes del entorno avícola, incluida la humedad, la temperatura, la propiedad del clima y el mando del ventilador. (Avicultura.info 2018)



Figura 2.12 Avicultura y el Internet de las cosas
Fuente: (Avicultura.info 2018)

2.5.7. IoT (Internet de las Cosas)

El Internet de las cosas proporciona energía a objetos (como comunicadores, cámaras, sensores, etc.) que anteriormente estaban conectados a través de circuitos cerrados y les conciernen informar globalmente mediante el uso de redes de red.

Si tenemos que definir Internet de las cosas, es mejor decir que se trata de una red que utiliza Internet para interconectar objetos físicos. Los objetos mencionados anteriormente utilizan un sistema integrado o el mismo hardware dedicado, que no solo reconoce la conexión a Internet, sino que además puede programar eventos concretos establecidos en tareas dirigidas de carácter remota. (Red Hat 2020)



Figura 2.13 (Internet de las Cosas)
Fuente: (copiasdeseguridadenlanube.com, 2021)

2.5.8. Su Funcionalidad

Como ya hemos dicho, todos estos trucos están en sistemas embebidos. En comparación con los teléfonos inteligentes, estos chips y circuitos parecen muy rudimentarios, pero tienen todas las herramientas necesarias para realizar tareas profesionales muy especiales. (Torres, 2014)

No hay ningún tipo de objeto específico conectado a Internet de las cosas. En cambio, se pueden catalogar en cosas que actúan como sensores y objetos que realizan operaciones activas. (Torres, 2014)

Sin embargo, la iniciación es igual, la clave es la maniobra remota. Cada objeto acoplado a Internet tiene una IP concreta, a través de la cual se puede consentir para acoger conocimientos. Del mismo modo, puede colocar en unión con un servidor externo y despachar sus datos recopilados. (Torres, 2014)

Poco a poco la IoT está desempeñándose en varios ámbitos de la siguiente manera:

- **Industria de producción en masa:** Máquina que controla el proceso de producción, robots ensambladores, sensores de temperatura, control de producción, cada vez más empresas están conectando todo a Internet, donde se puede controlar la infraestructura. (Torres, 2014)

- **Control de la infraestructura de la ciudad:** inspección de semáforos, viaductos, vías de tren, cámaras de la ciudad. En cada etapa son más las ciudades que están implementando este ejemplo de infraestructura basada en el Internet de las Cosas, de manera que se pueda monitorear el correcto funcionamiento de sus estructuras y sus operaciones más flexibles para adaptarse a los nuevos eventos. (Torres, 2014)
- **Sector sanitario:** La mayoría de las clínicas y hospitales de todo el mundo entregan a los sistemas que permiten a los profesionales sanitarios realizar un seguimiento activo de los pacientes de forma ambulatoria y no invasiva. (Torres, 2014)

2.6. Sensores y actuadores para IoT

Los sensores son dispositivos electrónicos que debe constar acoplado a una red de fundamentos. Son herramientas que se pueden utilizar para medir propiedades físicas y convertir dicha información en señales eléctricas u ópticas. (Tuatara Tech, 2021)

2.7. Tipos de Sensores

2.7.1. Sensor DHT11

Permite usar Arduino ya que puede calcular temperatura y humedad. Además de calcular la temperatura y la humedad, una de las ventajas que brinda el DHT11 es que es digital. (Del Valle,2021)

En la versión sin PCB hay 4 pines y en la versión con PCB se encuentran 3 pines.

DHT11 sin PCB y sus pines:

- VCC: alimentación
- I/O: transmisión de datos
- NC: no conecta, pin al aire
- GND: conexión a tierra

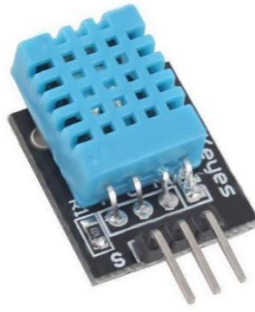


Figura 2.14 Sensor DHT11
Fuente: (Programarfacil.com 2021)

DHT11 con PCB y sus pines:

- GND: conexión con tierra
- DATA: transmisión de datos
- VCC: alimentación

Tabla 2.2 Características técnicas del DHT11

MODELO	DHT11
Alimentación	de 3,5 V a 5 V
Consumo	2,5 mA
Señal de salida	Digital
Temperatura	
Rango	de 0°C a 50°C
Precisión	a 25°C \pm 2°C
Resolución	1°C (8-bit)
Humedad	
Rango	de 20% RH a 90% RH
Precisión	entre 0°C y 50°C \pm 5% RH
Resolución	1% RH

Fuente: (Programarfacil.com 2021)

2.7.2. Sensor Humedad de suelo

Dispositivo proporcionado con un sensor de humedad de la superficie capaz de calcular la temperatura y la humedad de la superficie. Está ubicado en el extremo del cable con una carcasa impenetrable, y el interior está protegido por resina. El sensor tiene tres componentes de acero en contacto con materiales geológicos. (IOTSENSCity, 2021)

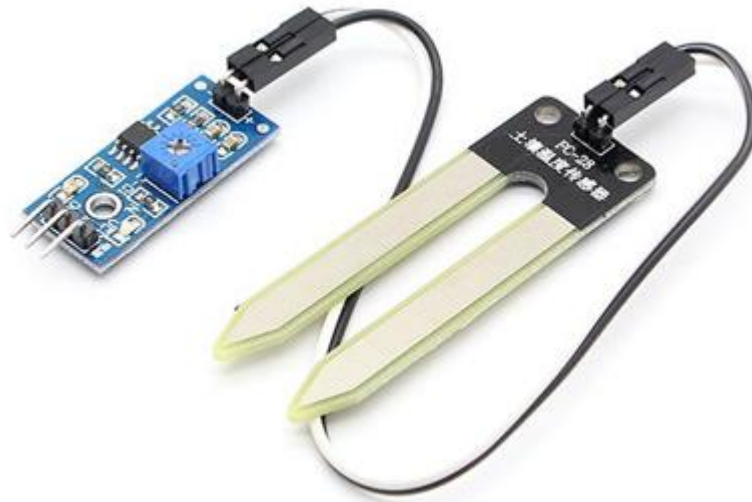


Figura 2.15 Sensor humedad de suelo
Fuente: (Luisllamas.es 2021)

- **Especificaciones Técnicas:**

- Voltaje de alimentación: 3.3V - 5V DC (VCC)
- Corriente de operación: 35mA
- Voltaje de señal de salida analógico (AO): 0 a VCC
- Voltaje de señal de salida digital (DO): 3.3V/5V TTL
- Opamp LM393 en modo comparador, umbral (threshold) regulable por potenciómetro
- Superficie de electrodo: Estaño
- Incluye: Electrodo, Placa y cable de conexión
- Vida útil electrodo sumergido: 3 a 6 meses
- Dimensiones YL-38: 30*16 mm
- Dimensiones YL-69: 60*20*5 mm

- **Conexiones:**

- **VCC:** Voltaje de alimentación (3.3V - 5V DC)
- **GND:** Tierra (GND 0V)
- **DO:** Salida digital
- **AO:** Salida analógica

2.7.3. Sensor intensidad de luz (luxómetro)

El sensor de luminosidad TSL2591 es un sensor de luz digital avanzado, perfecto para ser usado en una extensa gama de circunstancias de iluminación. Este sensor es tremendamente preciso, lo que aprueba cálculos exactos en lux y se logra configurar para otras ganancias para descubrir rangos de luz de 188 μ Lux hasta 88.000 Lux sobre la marcha.

La mejor integrante de este sensor es que contiene dos diodos, uno para sombra ultrarrojo y otro para aparición completo. Eso significa que usted puede cronometrar por divergente infrarrojo, espantajo acabado o luz patente por el ser humano. La universalismo de los sensores sólo pueden detectar una u otra, y esto no representa con desventaja lo que ven los ojos humanos (dado que no podemos recibir la luz infrarroja que es detectada por la conjunto de los fotodiodos). Este sensor es muy comparable a la TSL2561, no obstante, con un cargo más inclito. Este sensor tiene un enorme cargo realista de 600,000,000:1. A desajuste del TSL2561 no se puede transformarse la autoridad I2C, así que tenlo en cuenta.

Especificaciones técnicas:

- TSL2591 Hoja de datos
- Aproxima respuesta del ojo humano
- Rango dinámico extremadamente amplio del 1 al 600.000.000 Counts
- Lux Rango: 188 μ Lux hasta 88.000 Lux.
- Rango de temperatura: -30 a 80 °C
- Rango de tensión: 3.3-5V (Regulador integrado)
- Interfaz: I2C
- Este módulo utiliza una dirección fija I2C 7 bits 0x29
- Dimensiones: 19mm x 16mm x 1mm / .75" x .63" x .04"
- Peso: 1,1 g

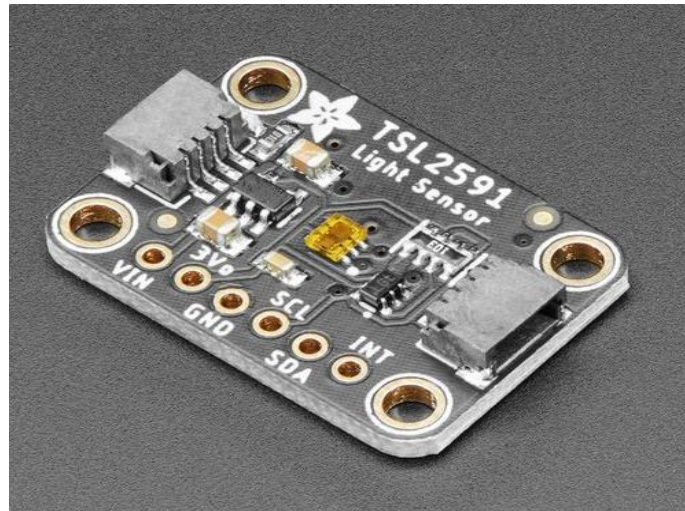


Figura 2.16 Sensor luxómetro
Fuente: (Electronilab.co 2021)

2.8. Actuadores

El actuador es un conector que puede convertir energía durante la activación del transcurso para tener un impacto en el proceso automatizado. Por ejemplo, en base a un conjunto de instrucciones específico, se puede usar para mover o controlar el motor básico de un mecanismo o sistema. (TUATARA TECH, 2020)

Los actuadores pueden hacer una función física, en otras palabras, realizar que las cosas sucedan. Un variedad de actuador es una llave, utilizada para controlar el sistema hidráulico. Hay tres tipos de actuadores utilizados en la IoT:

- Hidráulica: utiliza la boicoteo del líquido para realizar el acción maquinal, se emplean cuando se necesita brío.
- Neumáticos: utiliza aire pastilla a adhesión obstrucción para otorgar el funcionamiento maquinal.
- Eléctrico: impulsado por un motor que redime la fortaleza eléctrica para el funcionamiento automotriz.

2.9. Tipos de Actuadores

2.9.1. Actuador Hidráulico

Los actuadores hidráulicos son dispositivos automáticos que funcionan a quemarropa con aceite o agua. Son aptos para baja presión (250-500 psi) y entrada alta de presión (600-5000 psi). Utilizan agua de la planta de tratamiento, aceite hidráulico industrial y aceite biodegradable. Se utiliza para la automatización de válvulas de compuerta, válvulas de aeróstato, válvulas hombre, válvulas de mariposa o válvulas de amortiguación. (Instesista, 2021)



Figura 2.17 Actuador hidráulico
Fuente: (Conocimientolibre.mx 2021)

2.9.2. Características

- El tiempo de ejecución es más rápido que 1 segundo / pulgada. O menos. También están equipados con bombas hidráulicas manuales.
- Su uso correcto es en tuberías que trasladan líquidos o gases ácidos que no se pueden utilizar como fuerza motriz, lo que también requiere una revisión muy fina. Son magníficos para paradas de emergencia, segmentación de tuberías o contra sobretensiones.

2.9.3. Actuador Neumático

Los actuadores neumáticos manejan aire apretado como principio de energía, que es muy adecuado para inspeccionar movimientos rápidos, pero con claridad limitada. Entre ellos, la fuente de energía es el aire con una presión entre 5 y 10 bar (Intef, 2020).



Figura 2.18 Actuador neumático
Fuente: (Conocimientolibre.mx 2021)

2.9.4. Características

- Ángulo de Rotación y Pares de Maniobra Actuadores de aluminio con giro 0°-90° o 0°-180° para automatizar válvulas de bola, mariposa o macho cónico.
- Presión Consiguen ser maniobrados con aire, agua o fluido no agresivo hasta 8 bar de presión.
- Temperatura normal de funcionamiento De -32°C a +80°C Bajo demanda: - Baja temperatura (-55°C a + 80°C) -Alta temp. (-20°C a +120°C)

2.9.5. Actuador Eléctrico

Las particularidades de control, sencillez y exactitud de los accionamientos eléctricos han dispuesto que sean los más consumidos en los robots industriales actuales. (Intef, 2020)

Entre los actuadores eléctricos logran diferenciarse tres tipos diferentes:

- Motores de corriente continua (DC). Servomotores
- Motores paso a paso
- Motores de corriente alterna (AC)



Figura 2.19 Actuador eléctrico
Fuente: (Conocimientolibre.mx 2021)

2.10. Actuadores básicos de IoT

Los actuadores cambian una señal eléctrica en un aumento físico conveniente, como corriente, fuerza, sonido, etc. Un actuador también se clasifica como transductor porque cambia un tipo de cantidad física en otro y generalmente es activado u operado por una señal de comando de bajo voltaje. Los actuadores pueden clasificarse como dispositivos binarios o continuos según el número de estados estables que tenga su salida. (iotbytes.wordpress.com 2021)

Por ejemplo, un relé es un actuador binario ya que tiene dos estados estables, ya sea energizado y enclavado o desenergizado y desenganchado, mientras que un motor es un actuador continuo porque puede girar 360 ° completos.

Exploremos algunos de los actuadores básicos que puede utilizar en sus proyectos de IoT:

2.10.1. Servomotores

Servo es un pequeño dispositivo que contiene un motor de CC de dos cables, un tren de engranajes, un potenciómetro, un circuito integrado y un eje (columna de salida). Al enviar una señal codificada al servo sistema, el eje se puede colocar en una posición angular específica. Entre los tres cables que se extienden fuera de la carcasa del servo, uno es el cable de alimentación, uno es el cable de tierra y el otro

es el cable de entrada de control.

Cuando se aplica una señal de control al servo que representa la posición de salida deseada del servo eje, el servo (servo) suministrará energía a su motor de CC hasta que su eje gire a esa posición. Utiliza un dispositivo de detección de posición para determinar la posición de rotación del eje, por lo que sabe en qué dirección debe girar el motor para mover el eje a la posición ordenada.



Figura 2.20 Servomotores
Fuente: (iotbytes.wordpress.com 2021)

2.10.2. Motores paso a paso

Un motor paso a paso es un motor de CC que se mueve en pasos discretos. Tienen múltiples bobinas, que se organizan en grupos denominados "fases". Al energizar cada fase por turno, el motor girará, un paso a la vez. Mediante pasos controlados por computadora, puede lograr un posicionamiento y / o control de velocidad muy precisos.

El servomotor consume energía cuando gira a la posición ordenada, pero luego el servomotor se detiene. El motor paso a paso sigue consumiendo energía para bloquear y mantener una posición ordenada.



Figura 2.21 Motores paso a paso
Fuente: (iotbytes.wordpress.com 2021)

2.10.3. Motores de CC (motores de rotación continua)

Estos motores de corriente continua (CC) son los actuadores más utilizados en proyectos electrónicos. Son simples, económicos y fáciles de usar. Además, están disponibles en una variedad de tamaños para adaptarse a diferentes tareas. Los motores de CC convierten la energía eléctrica en energía mecánica. Constan de imanes permanentes y bobinas internas. Cuando se aplica corriente, los bucles de alambre generan un campo magnético, que reacciona contra el campo exterior de los imanes estáticos.



Figura 2.22 Motor de CC
Fuente: (iotbytes.wordpress.com 2021)

2.10.4. Actuador lineal

Un actuador lineal es un actuador que produce un movimiento lineal, que es diferente del movimiento circular de un motor tradicional. Los actuadores lineales se utilizan en máquinas herramienta y maquinaria industrial, periféricos de computadora (como variadores e impresoras), válvulas y amortiguadores, y muchos otros lugares donde se requiere movimiento lineal.



Figura 2.23 Actuador lineal
Fuente: (iotbytes.wordpress.com 2021)

2.10.5. Relevó

Un relé es un interruptor eléctrico. Muchos relés utilizan electroimanes para operar interruptores mecánicamente, pero también se utilizan otros principios de funcionamiento, como los relés de estado sólido. La ventaja del relé es que la potencia requerida para operar la bobina del relé es relativamente pequeña, pero el relé en sí puede usarse para controlar motores, calentadores, luces o circuitos de CA que consumen más energía.



Figura 2.24 Relevó
Fuente: (iotbytes.wordpress.com 2021)

2.10.6. Selenoide

Un solenoide es simplemente un electroimán especialmente diseñado. Los solenoides son económicos y su uso se limita principalmente a aplicaciones de encendido y apagado, como enclavamiento, bloqueo y activación. Se utilizan con frecuencia en electrodomésticos (por ejemplo, válvulas de lavadoras), equipo de oficina (por ejemplo, fotocopiadoras), automóviles (por ejemplo, pestillos de puertas y solenoide de arranque), máquinas de pinball (por ejemplo, émbolos y parachoques) y automatización de fábricas.



Figura 2.25 Selenoide
Fuente: (iotbytes.wordpress.com 2021)

2.11. ESP32

El módulo ESP32 es una máquina todo en uno certificada Wi-Fi / Bluetooth. No solo proporciona transmisión inalámbrica, sino que también posee un procesador e interfaz integrados con el fin de conectar varios dispositivos periféricos. Este procesador contiene 2 núcleos de procesamiento, y sus frecuencias operativas se pueden controlar de forma independiente entre 80 (MHz) y 240 MHz.

La plataforma ESP32 brinda a costo accesible capacidades wifi, bluetooth y BLE. Posee un CPU con dos núcleos de hasta 240Mhz que pueden ser controlados independientemente. Además, está constituido internamente por una gran cantidad de periféricos conteniendo: sensores táctiles capacitivos, sensor de efecto Hall, amplificadores de bajo ruido, interfaz para tarjeta SD, Ethernet, SPI de alta velocidad, UART, I2S e I2C.

El lugar fuerte de esta plataforma es que posee una gran comunidad en internet que le ofrece soporte y desarrolla incesantemente nuevos equipos para su consumo.

Para su proceso cuenta con gran diversidad de software, lenguajes de programación, frameworks, librerías, códigos, y otros recursos. Los más comunes a elegir son: Espressif IoT Development Framework (Espressif IoT Development Framework) desarrollado por el fabricante del chip, Arduino (en lenguaje C++), Simba Embedded Programming Platform (en lenguaje Python), RTOS's (como Zephyr Project, Mongoose OS, NuttX RTOS), MicroPython, LUA, JavaScript (Espruino, Duktape, Mongoose JS), Basic (espressif.com 2020)



Figura 2.26 ESP32
Fuente: (espressif.com 2020)

2.11.1. Características Técnicas del ESP32

- Procesador Tensilica Xtensa 32bits LX6 hasta 240MHz
- Wi-Fi: 802.11b/g/n/e/i (802.11n @ 2.4 GHz hasta 150 Mbit/s)
- Bluetooth: v4.2 BR/EDR y Bluetooth Low Energy (BLE)
- Rom:448 KiB
- SRAM: 520 KiB
- RTC slow SRAM: 8 KiB
- RTC fast SRAM: 8 KiB
- eFuse: 1 kibit
- Flash embebida: 0 MiB (ESP32-D0WDQ6, ESP32-D0WD, and ESP32-S0WD chips); 2 MiB (ESP32-D2WD chip); 4 MiB (ESP32-PICO-D4 SIP module)
- Periféricos compatibles: ADC, DAC, I2C, UART, Interfaz CAN 2.0, SPI, I2S, RMI y PWM entre otros.
- Seguridad tipo IEEE 802.11, WPA, WPA/WPA2 y WAPI
- Memoria SRAM de 520kBytes
- Dispone de 48 Pines

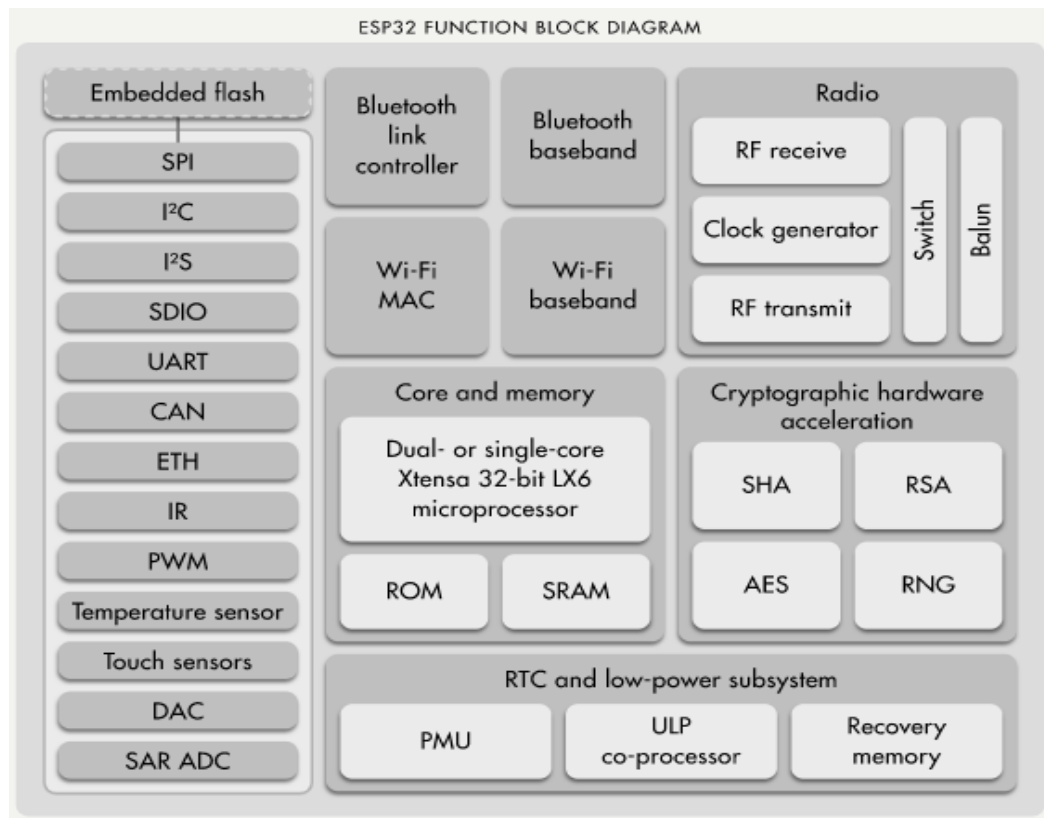


Figura 2.27 Diagrama de bloque funcionamiento ESP32
Fuente: (espressif.com 2020)

2.12. Servicio web en la nube

La nube posee de todas las cuentas de Internet que pueden guardar información: correo electrónico, documentos en línea, cuentas de sitios de radio por Internet, etc. Uno de los modelos más sencillos de computación en la nube son los programas, como Google Documents. Al establecer una cuenta, debido a que la información se almacena en el servidor de Documentos de Google y no en la computadora, se puede acceder a estos documentos desde cualquier computadora con conexión a Internet. Google Docs es la única aplicación basada en la nube. (Techlandia.com 2021)



Figura 2.28 Cloud computing
Fuente: (Xiaomitoday.com 2021)

2.12.1. Beneficios de la nube

La dirección es el mayor beneficio de usar aplicaciones en la nube. La mayoría de los bosquejos existentes son muy simples, pero muy seguros: cualquiera obtiene iniciar sesión siempre que tenga una cuenta y una contraseña. Esto significa que los empleados ya no están restringidos a sus computadoras, porque con los servidores en la nube, pueden permitir a sus documentos en cualquier otra computadora. (Techlandia.com 2021)

2.12.2. Potencial de la nube

Imagínese que está descubierto en un aeropuerto, con los escritos importantes para una próxima reunión. Consigue colgar estos documentos en el servidor en la nube y compartirlos con sus colegas donde se podrá charlar con ellos durante la reunión incluso estando aun en el aeropuerto. (Techlandia.com 2021)

2.12.3. Hardware

De hecho, los servidores en la nube no son tan pesados como los que se utilizan en los servicios acostumbrados a los consumidores. Dichos servidores no se utilizan para almacenar y aprobar a documentos (la información permanece en la base de datos). En cambio, cualesquiera de los servidores deben ejecutar un sitio web. La ligereza de la computación en nube también abarata los servidores porque utilizan menos hardware. (Techlandia.com 2021)

2.12.4. Importancia de la nube

Los servidores en la nube superan las barreras de los sistemas operativos. Los usuarios de Mac permiten a la misma Internet que los usuarios de PC. Dado que la coincidencia del software no es un problema, esto forja que la nube exista una introducción extremadamente importante. (Techlandia.com 2021)

2.13. AWS (Amazon Web Services)

Amazon Web Services (AWS) plataforma de actividades comerciales en la nube que suministra diferentes servicios de productos, como almacenamiento, red, plataforma de datos, productos de aplicaciones, potencia de cómputo, mensajería, inteligencia artificial, servicios móviles, seguridad, identidad, etc. Permitiendo que la empresa crezca.

Amazon Web Services con el acrónimo AWS se ha convertido en una plataforma en la nube muy popular y completa debido a los diversos servicios que brinda. Muchas empresas están creciendo a un ritmo vertiginoso, lo que permite responder con la operación de su infraestructura, aumentar así la productividad y reducir los costos. (Amazon Web Services 2020)



Figura 2.29 AWS (Amazon Web Services)
Fuente: (Hiberus.com 2020)

2.13.1. ¿Para qué sirve Amazon Web Services?

Amazon Web Services proporciona una amplia variedad de servicios que pueden resolver diversas variables en la infraestructura de TI.

AWS nos proporciona una plataforma más segura y confiable, lo que permite a los arquitectos de TI encontrar todos los recursos necesarios a través de Amazon para ensamblar su arquitectura completa, segura y tolerante a fallas.

2.13.2. Beneficios

La mayor ventaja que Amazon nos puede brindar con aliados es que nos garantiza que sólo nos enfocamos en lo que se debe hacer, es decir, para los proveedores de desarrollo de software, el foco principal es desarrollar software. (Robledano, 2019)

Se acabó la dependencia del "quién" que configura o implementa una infraestructura determinada. Para los proveedores de nube como AWS, el campo de la administración de sistemas está comprometido con administrar los recursos técnicos de una manera más optimizada y sencilla, de manera de brindarlos en sus plataformas. 99,9% de estabilidad en su servicio o almacenamiento. (Amazon Web Services 2020)

2.13.3. Herramientas/Servicios

- **Amazon EC2:** Proporciona todo lo necesario para administrar para la gestión y administración de instancias.
- **Amazon RDS:** Provee servicios de base de datos, entre ellos podemos elegir los motores de base de datos más comunes, como MYSQL, PostgreSQL, SQL SERVER, Oracle, Amazon Aurora y NoSQL.
- **Amazon VPC:** Brinda servicios para la creación de redes y subredes a través de su servicio llamado Amazon VPC.
- **Amazon S3:** Como su nombre lo indica, el almacenamiento de archivos nos permite almacenar varios objetos, y sus tipos de acceso se clasifican en acceso regular, poco frecuente o muy frecuente.
- **IAM:** Otorga el control de acceso, que está relacionado con la autenticación

de usuarios, básicamente pueden ayudarnos a administrar el acceso a los recursos.

2.13.4. Herramientas Administrativas

- **Amazon Cloudwatch-** Esta es una herramienta de monitoreo que da el acceso a rastrear métricas y registros de los servicios.
- **Amazon CloudFormation-** Proporciona servicios para crear y administrar colecciones de recursos, configurar y actualizar recursos plantillas escritas en Json.
- **Amazon CloudTrail-** Se encarga de la auditoría de API a través de servicios web.

2.14. AWS IoT

AWS posee una gran variedad de poderosos productos de IoT desde sitios de borde hasta la nube. AWS IoT se podría decir que es uno de los único que provee en la nube y que combina análisis y administración de datos enriquecidos en servicios fáciles de manejar y diseñados para datos ruidosos de IoT. (Hiberus.com 2020)

2.14.1. ¿Qué tipos de servicios ofrece AWS IoT?

AWS IoT proporciona actividades comerciales para todas las capas de seguridad, incluyendo los dispositivos de seguridad preventiva (como el control de acceso y el cifrado de datos del dispositivo) y una asistencia para la supervisión y auditoría continuas de las configuraciones. (Hiberus.com 2020)

2.14.2. Casos de uso de la AWS IoT

- **Industria:** Los usuarios de AWS IoT planifican diseños de aplicaciones de IoT industriales para garantizar el sustento predictivo y control de calidad, y para poder monitorear las instrucciones de forma remota.
- **Casa conectada:** Los usuarios de AWS IoT están planteando aplicaciones de redes domésticas, seguridad, vigilancia y automatización del hogar conectado.
- **Comercial:** Los usuarios de AWS IoT están creando aplicaciones comerciales para la supervisión de la salud y el tráfico y la seguridad pública.

2.15. TimeStream

Amazon Timestream es un servicio de conjuntos de datos de series de tiempo rápido, escalable y sin asistente para aplicaciones operativas y de IoT. Le permite almacenar y analizar fácilmente billones de eventos por día a una velocidad 1000 veces mayor que el costo. Amazon Timestream le ahorra el tiempo y el costo de administrar el periodo de vida de las reseñas de series de tiempo al almacenar los antecedentes más nuevos en la memoria y mover los datos antiguos a una capa de provisión con costos optimizados según las estrategias definidas por el nombre del usuario

El motor de consultas personalizadas de Amazon Timestream le permite acceder y analizar datos combinados nuevos y antiguos sin tener que detallar abiertamente en la sugerencia si los datos ocupan en la memoria o en una capa de costo optimizado. (Amazon Timestream, 2020)

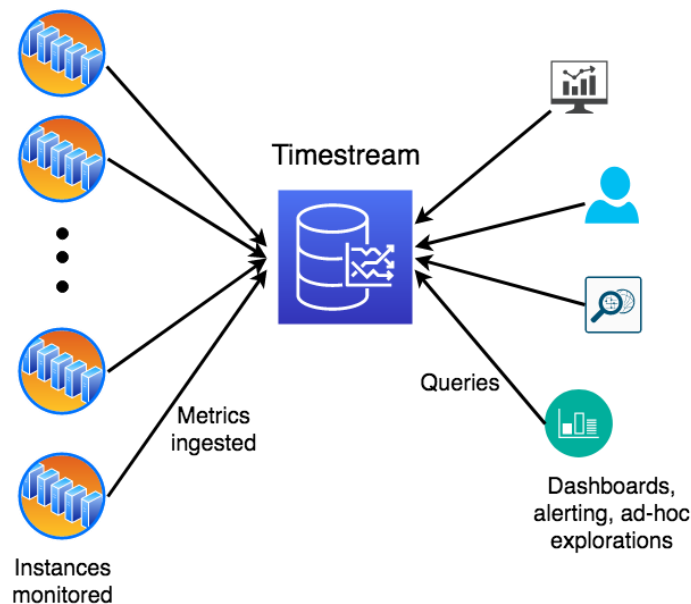


Figura 2.30 Amazon Timestream
Fuente: (aws.amazon.com 2021)

2.15.1. Beneficios

- Amazon Timestream le ofrece la escalera para procesar billones de eventos y millones de consultas por día. A metropolitano que su constancia necesite progresar, se escalera rápidamente para acompasar la capacidad.
- El motor de consultas especialmente diseñado de Amazon Timestream aceptar de forma inmaculado y combina datos en distintas capas de almacenamiento sin que usted tenga que puntualizar la localización de los datos.

2.16. Amazon DynamoDB

Amazon DynamoDB Es una conjunto de datos multirregional duradera y totalmente administrada con señal, recuperación y calma integrados, de igual modo una memoria caché para concentraciones a escala de Internet. DynamoDB puede guiar más de 10 billones de solicitudes por día y puede sufrir un zapapico de más de 20 millones de solicitudes por santiamén. Centenas de miles de consumidores de AWS eligen DynamoDB a manera su valor clave y conjunto de datos de documentos para dispositivos móviles, web, de juegos, tecnología promocional y de IoT, así como más aplicaciones que necesitan obtener a datos de despreciable latencia de cualquier escala.



Figura 2.31 DynamoDB
Fuente: (Aws.amazon.com 2021b)

2.16.1. Beneficios

DynamoDB admite algunas de las aplicaciones más grandes del comunidad y proporciona un legislatura de respuesta de milisegundos de un solo dígito de cualquier tamaño. Puede producir aplicaciones con aptitud de procesamiento y almacenaje casi ilimitada. DynamoDB popular Tables alega sus datos en varias regiones de AWS, lo que le permite penetrar rápida y localmente a los datos desde aplicaciones distribuidas globalmente. Para casos de uso que requieren un paso más veloz con latencia de microsegundos, DynamoDB Accelerator (DAX) proporciona una caché en memoria completamente administrada.

DynamoDB expandirá o encogerá automáticamente la mesa hacia concordar el aforo y mantener el beneficio. Funciones integradas de usabilidad y tolerancia a fallos, por lo que no necesita considerar estas funciones al diseñar su aplicación. DynamoDB proporciona contenido bajo petición y modelos de suministro de capacidad para que pueda optimizar los costos especificando la contenido por carga de trabajo o pagando por los recursos que consume.

2.17. Descripción de las configuraciones en AWS

En la figura 4.17 se observa un esquemático de conexiones con servicios de AWS, entre ellos se encuentra AWS IoT, Amazon Kinesis, Lambda, DynamoDB y Amazon S3 entre los más importantes.

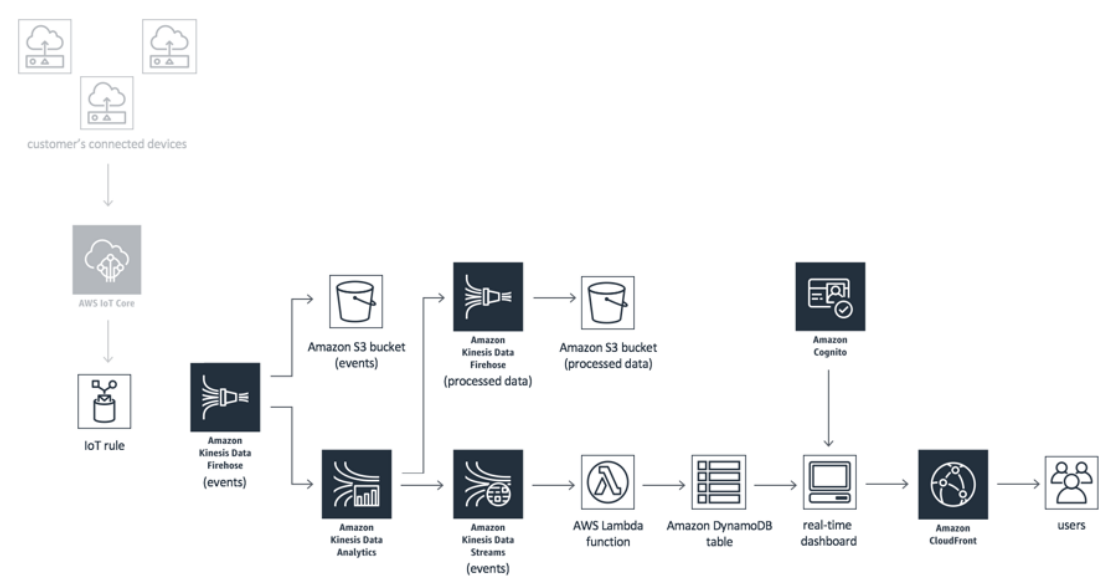


Figura 2.32 Componentes y funcionalidad de mediante la utilización de Real-Time IoT Device Monitoring con Kinesis Data Analytics
Fuente: (Aws.amazon.com 2021a)

La plantilla de AWS CloudFormation implementa una regla de AWS IoT, dos flujos de entrega de Amazon Kinesis Data Firehose, buckets de Amazon Simple Storage Service (Amazon S3), una aplicación de análisis de datos de Kinesis, una función de AWS Lambda, una tabla de Amazon DynamoDB y grupos de usuarios de Amazon Cognito. Las distribuciones de Amazon CloudFront y los paneles de control de dispositivos en tiempo real se utilizan para leer y mostrar de forma segura la conexión del dispositivo y las métricas de actividad almacenadas en tablas de DynamoDB.

Cuando AWS IoT extrae datos de su dispositivo conectado, las reglas de AWS IoT envían los datos al flujo de transferencia de datos de Kinesis. El flujo de entrega archiva eventos en buckets de Amazon S3 y envía los datos a la aplicación de análisis de datos de Kinesis para su procesamiento. La aplicación envía los datos procesados a la función Lambda, que los envía en tiempo real a la tabla de DynamoDB para ser almacenados. La aplicación también envía los datos procesados al segundo flujo de entrega de datos de Kinesis, que los archiva en un bucket de Amazon S3.

La solución también creó un grupo de usuarios de Amazon Cognito, una distribución de Amazon CloudFront y un panel en tiempo real alojado en un depósito de Amazon

S3 para leer y mostrar de forma segura la actividad del dispositivo almacenada en las tablas de DynamoDB (Aws.amazon.com 2021a).

2.17.1. Protocolo MQTT

El protocolo MQTT utiliza estándares de información de publicación / suscripción para sus hosts. Para enviar los datos, el dispositivo publica un mensaje para identificar el tema del agente de mensajes de AWS IoT. Para recibir mensajes del intermediario de mensajes, los dispositivos determinan qué tema elegirán enviando filtros de temas al intermediario de mensajes en una solicitud de suscripción. El motor de reglas de AWS IoT recibe mensajes MQTT del agente de mensajes (Aws.amazon.com 2021a).

2.17.2. Reglas de AWS IoT

Las reglas de AWS IoT constan de declaraciones de consulta de reglas y una o más acciones de reglas. Cuando el motor de reglas de AWS IoT recibe un mensaje MQTT, estos elementos realizan las siguientes operaciones en el mensaje.

- **Instrucción de consulta de reglas:** La declaración de consulta normal describe el tema MQTT que se utilizará, interpreta los datos de carga en el mensaje y formatea los datos de la manera descrita en la declaración SQL, que es similar a la declaración utilizada por la base de datos. Datos SQL populares. El resultado de la declaración de consulta son los datos enviados a la operación de la regla
- **Acción de la regla:** Cada acción de regla en la regla actúa sobre los datos generados por la instrucción de consulta de regla. Admite AWS IoT T. Sin embargo, en este tutorial, se centrará en la reedición, que publica los resultados de las declaraciones de consulta como mensajes MQTT específicos de un tema.

2.17.3. Temas MQTT

Los clientes de AWS IoT identifican sus mensajes publicados proporcionando el nombre del asunto del mensaje. Los clientes determinan a qué mensajes quieren suscribirse (recibir) registrando filtros de tema en AWS IoT Core. El intermediario de mensajes utiliza filtros y nombres de temas para enrutar los mensajes desde los clientes de publicación a los clientes suscriptores. El intermediario de mensajes usa el asunto para identificar el mensaje enviado usando MQTT y la URL del mensaje HTTPS enviado usando HTTP.

AWS IoT utiliza temas para identificar los mensajes recibidos de los clientes de publicación y seleccionar los mensajes que se enviarán a los clientes suscriptores.

2.17.4. Nombres de temas

El nombre del sujeto y el filtro del sujeto son cadenas codificadas en UTF-8. Pueden representar jerarquías de información mediante el uso de caracteres de barra inclinada (/) para separar niveles en la jerarquía. Por ejemplo, el nombre del sujeto puede hacer referencia al sensor de temperatura en la habitación 1:

- `sensor/temperature/room1`

Al considerar el nombre del asunto de los mensajes del sistema, tenga en cuenta:

- El nombre del tema y el filtro de tema distinguen entre mayúsculas y minúsculas.
- El nombre del tema no debe disponer de información de identificación personal.
- Los nombres de los temas que comienzan con \$ son temas reservados para uso exclusivo de AWS IoT Core.
- AWS IoT Core no puede enviar ni recibir mensajes entre cuentas o regiones de AWS.

2.17.5. Filtros de temas

Los clientes de suscripción registran filtros de temas con el intermediario de mensajes para especificar los temas de los mensajes que el intermediario de mensajes debe enviarles. El filtro de tema puede ser un solo nombre de tema para suscribirse a un solo nombre de tema, o puede contener comodines para suscribirse a varios nombres de tema al mismo tiempo. (Aws.amazon.com 2021a).

El cliente de publicación no puede utilizar comodines en el nombre de su tema de publicación.

Uso de comodines con el ejemplo de nombre de tema del sensor habitación 1:

- Una suscripción a `sensor/#` recibe los mensajes publicados en `sensor/`, `sensor/temperature` y `sensor/temperature/room1`, pero no los mensajes publicados en `Sensor`.
- Una suscripción a `sensor+/room1` recibe mensajes publicados en `sensor/temperature/room1` y `sensor/humidity/room1`, pero no mensajes enviados a `sensor/temperature/room2`.

2.18. Grafana Cloud

Grafana Cloud es una plataforma de observabilidad componible que integra métricas, seguimiento y registros con Grafana. Aproveche el mejor software de observabilidad de código abierto, incluidos Prometheus, Loki y Tempo, sin la sobrecarga de instalar, mantener y expandir la pila de observabilidad (Grafana Cloud, 2021).

Grafana Cloud es una plataforma de observación escalable, de alto rendimiento y alta disponibilidad adecuada para sus aplicaciones e infraestructura. Proporciona una vista centralizada de todos los datos observables, ya sea que estén en el servicio Grafana Cloud Metrics o en su propio entorno de nube. Con soporte nativo para muchas fuentes de datos populares como Prometheus, Elasticsearch y Amazon CloudWatch, todo lo que necesita hacer para comenzar a crear paneles y consultar datos de métricas es configurar la fuente de datos en Grafana Cloud.



Figura 2.33 Grafana Cloud
Fuente: (Grafana.com 2021)

2.18.1. Características

Grafana Cloud se utiliza para cambiar sin problemas entre métricas, registros y seguimiento. Vincula datos y encuentra la causa raíz de forma más fácil y rápida. Reduzca el tiempo medio de recuperación (MTTR) e introduzca funciones de mitigación.

2.18.2. Integraciones en la nube

Los proyectos de código abierto de Grafana y Prometheus son estándares de observación de facto y son ampliamente adoptados por las bases. Ambos son fáciles de usar y empezar. Grafana Cloud tiene un conjunto en constante expansión de integraciones predefinidas para aprovechar al máximo estas tecnologías de forma rápida y sencilla. Con solo unos pocos clics, puede monitorear componentes de infraestructura populares como MySQL, Postgres, Redis o Memcache. Elija entre paneles y alertas preconfigurados, y use nuestro agente integral para recopilar métricas, registros y seguimiento importantes en un solo lugar.

2.18.3. Paneles de control en la nube (Grafana)

Grafana le permite consultar, ver, alertar y comprender sus métricas, sin importar dónde estén almacenadas. Cree, explore y comparta paneles con su equipo y cultive una cultura basada en datos.



Figura 2.34 Grafana Cloud
Fuente: (Grafana.com 2021)

3. Marco metodológico

3.1. Tipo de investigación

El proyecto se basa en una investigación exploratoria, buscando una visión general o amplia de la realidad, y su característica es la preparación.

Se realizó una investigación exploratoria debido a que la Universidad Politécnica Salesiana Sede Guayaquil no estudió las variables de investigación de manera práctica, por lo que es necesario analizar algunos aspectos relevantes del sistema de Internet de las Cosas utilizado para la avicultura. (Arturo et al. 2011)

Además, esto es experimental, porque ciertas cosas se verán afectadas positivamente al observar los resultados. (Ouali et al. 2016).

3.2. Metodología del trabajo

La metodología por utilizar es la experimental, para este proyecto se realizará la adquisición de datos mediante los sensores de humedad, temperatura y luz, los mismos que a través de un microcontrolador como el ESP32 e internet serán enviados a AWS IoT para ser procesados y gestionados de manera tal que el granjero pueda recibir alertas que sirvan en la toma de decisiones a la hora de una correcta crianza de aves de consumo.

Se tomó como bases anteriores proyectos de investigación vinculados como los que se cita a continuación:

3.2.1. Control y monitoreo de un criadero avícola controlado por microcontrolador desde un sitio web dinámico.

Autor: Isabel San Lucas, Diana Garzón
Universidad Politécnica Salesiana

En su tesis planteada "Control y monitoreo de un criadero avícola controlado por micro controlador desde un sitio web dinámico." Su objetivo es desarrollar un sistema automatizado que sea beneficioso para la industria avícola, optimizar las características de sus productos y ahorrar costos. Desempeñando de la siguiente manera: Al embolsar a la página www.sisavic.com presentará la pantalla auténtico del sistema, cobrar su nota de usuario y contraseña respectiva, amurmará una página personalizada, la misma que le autorizará monitorear y controlar su depósito de avícola: temperatura, sostenimiento, hidratación y acecho; Toda la automatización del sistema se ha realizado con un microprocesador, el mismo que

activa/desactiva electroválvulas de categoría solenoides para la orden del agua según el estado de los sensores. Para controlar la orden del manutención, en el depósito que contiene el balanceado se instaló un sensor, el cual envía la mojón a la que se encuentra el sustento balanceado en el silo; en los comederos se instalaron sensores que miden grado vasto y soez, todo es reflejado en el web site, si la temperatura afuera muy abyecta se enciende calefactor y envía una intranquilidad mediante e-mail al apoderado, de lo opuesto si afuera muy ingreso la temperatura se enciende un enfriador y envía una sorpresa mediante e-mail al administrador.

3.2.2. Control y monitorización de una granja avícola

Autor: Olmos Cabrera, Rubén
Universidad Politécnica de Madrid

El objetivo del proyecto que ofrece es proporcionar alternativas tecnológicas y de monitoreo urbano de bajo valor para granjas e invernaderos alimentados de forma independiente por sistemas fotovoltaicos. La solución se implementa en un modelo pequeño en el que se instalan los sensores y actuadores necesarios para la monitorización urbana.

Estos dispositivos se conectan mediante cables al microcontrolador Arduino Uno, que a su vez se conecta de forma inalámbrica al procesador a través de la tecnología Xbee. El procesador realiza las funciones de interfaz de usuario de monitoreo y mantenimiento de atmósfera de LabVIEWTM. El monitoreo de la señal se puede realizar en vivo o más tarde, porque los registros de los sensores se almacenan en la copia de seguridad de datos de MySQL

Se puede controlar desde cualquier punto de acceso conectado a un servidor web con archivos ejecutables de LabVIEWTM. [16]

3.2.3. Diseño e implementación de sistema de monitoreo automatizado en granja avícola

Autores: Gregorio Superior Castillo Quiroz, Arnulfo Cruz Garrido, Elisa Gonzaga Licon, Eugenio Luna Mejía

Revista de investigación en tecnologías de la información RITI

La adquisición de pollo en México, ha agrandado más o menos del 145% durante los últimos vida. Nuestra división de Huauchinango tiene una consecución destacado de engorda de pollos, el coetáneo artículo se basó en el desarrollo de un sistema de municipal automatizado de regulación de variables: temperatura, humedad, luz, pareja de flatulencia, amoniaco y el dióxido de carbono producidos por los desechos

en galeras.

Cuyo ampliación es obtener un crecimiento en el logro, optimizar las condiciones ambientales, reducir el índice de mortalidad de las aves mediante el diseño e integración de un sistema de patrulla automatizado contestando la prisa de facilitar a los productores avícolas a obtener un provecho óptimo de sus aves, que en su colectividad se realiza en manera manual.

3.3. Título de la propuesta

Diseño e implementación de un prototipo IoT para el monitoreo de parámetros ambientales aplicados a la avicultura para la crianza de pollos de granja utilizando hardware de bajo costo y AWS.

3.4. Justificación

La necesidad de evitar la mortalidad excesiva en la crianza de pollos de granja que acarrea pérdidas económicas para el pequeño avicultor incentiva a la investigación y realización de este prototipo IoT de monitoreo de medidas ambientales que influyan en el correcto progreso y crianza de las aves de corral.

Con la realización de este proyecto no solo se logrará una tecnificación para la avicultura en zonas rurales del Ecuador que mejore las condiciones técnicas para los pequeños avicultores, sino que también servirá como guía de aprendizaje para los futuros ingenieros electrónicos o de telecomunicaciones de la Universidad Politécnica Salesiana.

3.5. Beneficiarios de la propuesta

Este proyecto tendrá como beneficiarios directos a los pequeños avicultores de las zonas rurales de la costa ecuatoriana, ya que las pruebas se realizarán en el cantón Durán con parámetros ambientales del sector donde se pondrá en funcionamiento el prototipo IoT.

Además, se verán beneficiados los colegas de la carrera de ingeniería electrónica, puesto que se realizarán manuales informativos y de práctica sobre el prototipo, para futuros laboratorios y mejoras que ellos puedan hacer al trabajo.

3.6. Propuesta de solución

Se propone el diseño e implementación de un prototipo IoT para el monitoreo de parámetros ambientales aplicados a la avicultura para la crianza de pollos de granja

utilizando hardware de bajo costo y AWS.

Se muestra a continuación una gráfica de la propuesta planteada en este trabajo de investigación.

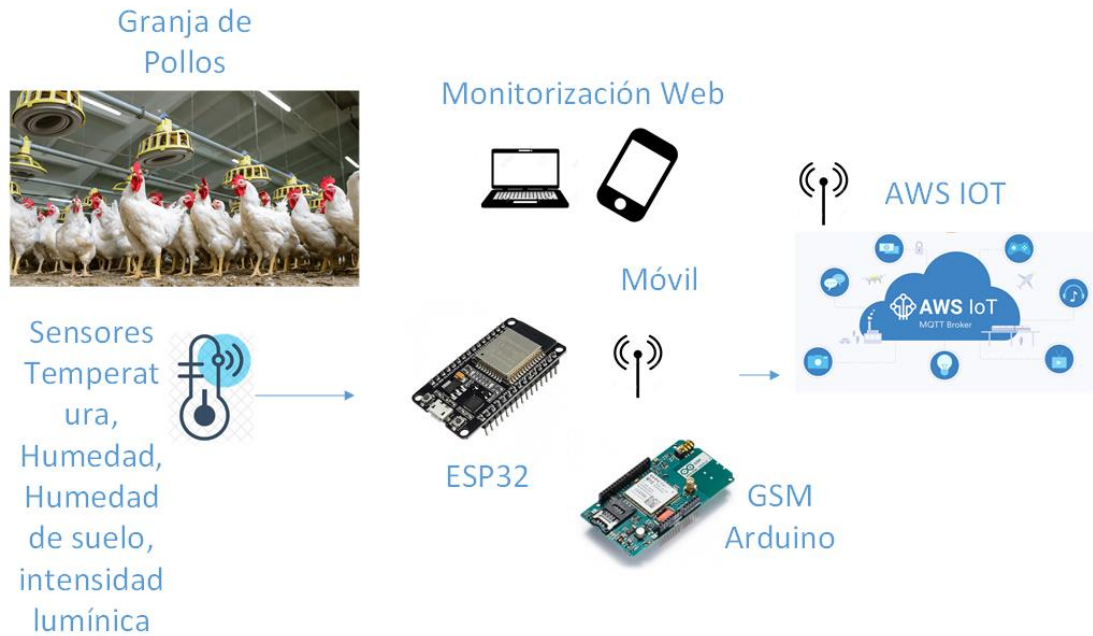


Figura 3.1 Prototipo IoT aplicado a la avicultura

Se muestra un esquemático de cuadros de las etapas del desarrollo del prototipo IoT aplicado a la avicultura.

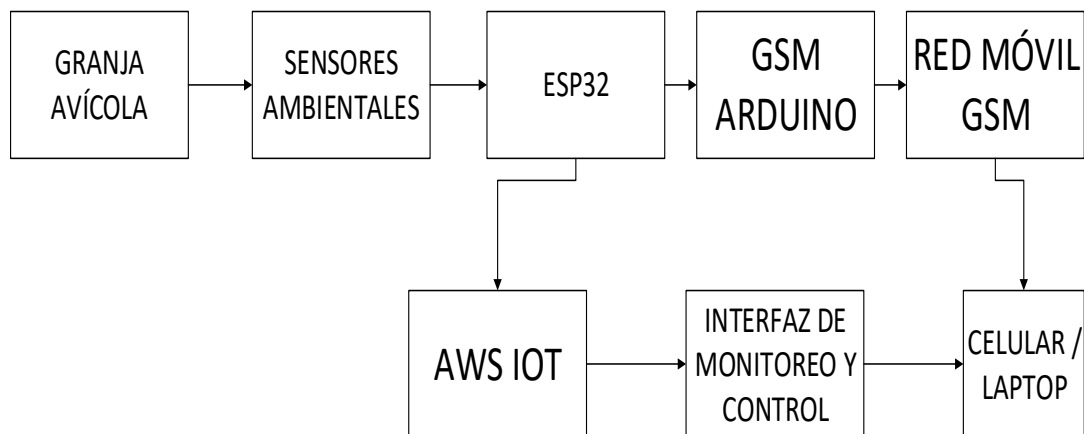


Figura 3.2 Esquemático del prototipo IoT aplicada a la avicultura

La monitorización de parámetros ambientales como temperatura y humedad ambiente, humedad de suelo e intensidad lumínica aplicados a granjas avícolas para la crianza de pollos de consumo se realizará con sensores de bajo costo y el microcontrolador ESP32.

Como se contempla en la figura 3.2, la propuesta es mediante el microcontrolador ESP32 el cual procesará todas las reseñas obtenidos mediante los sensores y serán enviados a través de una red inalámbrica al internet para que estos datos sean almacenados y tratados en servidores web en la nube como lo es AWS Amazon Web Services aplicado a IoT (internet de las cosas).

También se observa un módulo GSM de Arduino el cual trabaja con chip móvil para el envío de datos fuera del umbral de operación, con la finalidad de alertar mediante SMS al granjero y de esta manera tomar una decisión a la hora de criar pollos de granja.

La data almacenada y tratada por AWS IoT será mostrada mediante AWS IoT Core.

Los datos obtenidos por los sensores permitirán automatizar actuadores que activen ventiladores o control lumínico para que la decisión del control no solo sea manual por el avicultor sino también automatizada.

3.7. Diseño electrónico del prototipo IoT

Para el diseño del prototipo se realizaron los siguientes diseños esquemáticos electrónicos que se muestran posteriormente:

En la figura 3.3 se muestra el datasheet del componente ESP32 Wi-fi Core con el cual se trabaja para el prototipo de IoT aplicado a la agricultura.

ESP32 Wi-fi CORE

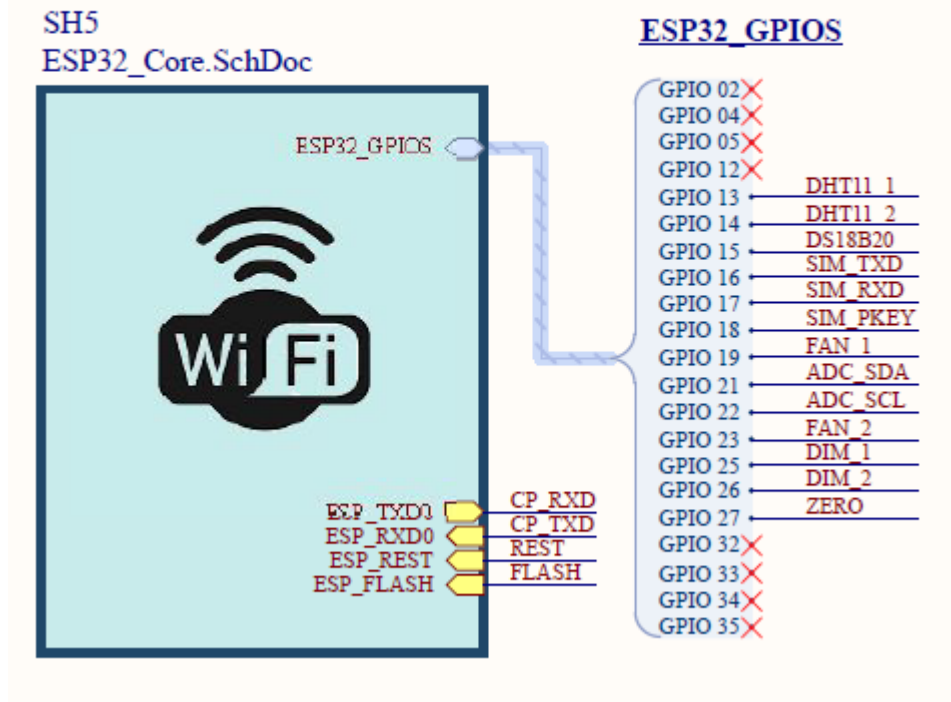


Figura 3.3 ESP32 Wifi Core

En la figura 3.4 se muestra el datasheet del componente USB To UART Bridge con el cual se trabaja para el prototipo de IoT aplicado a la agricultura, para convertir USB a Serial.

USB TO UART BRIDGE

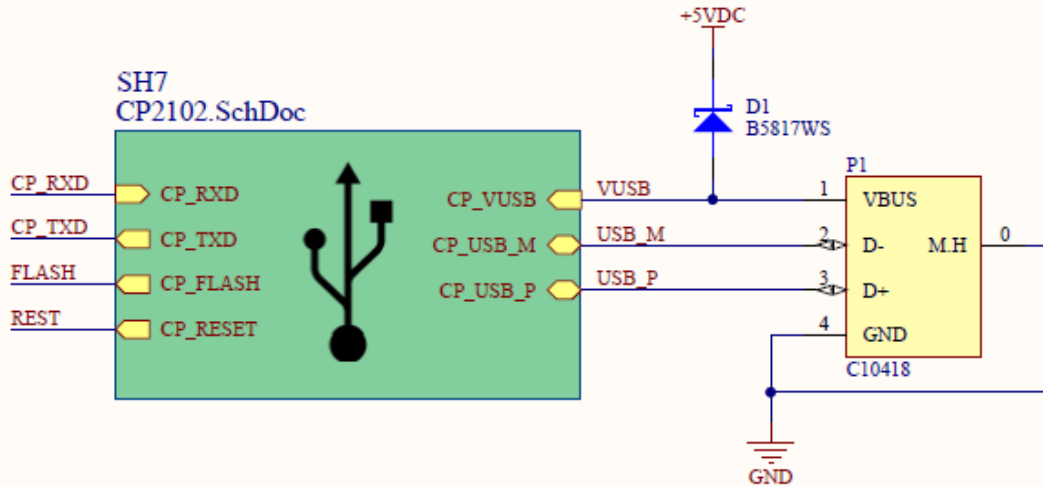


Figura 3.4 Convertidor USB a Serial

Se muestra en la figura 3.5 los diferentes tipos de reguladores y protección entre 5+, 4.1+ y 3.3 VDC, que se utilizan y la protección que tendrá el prototipo.

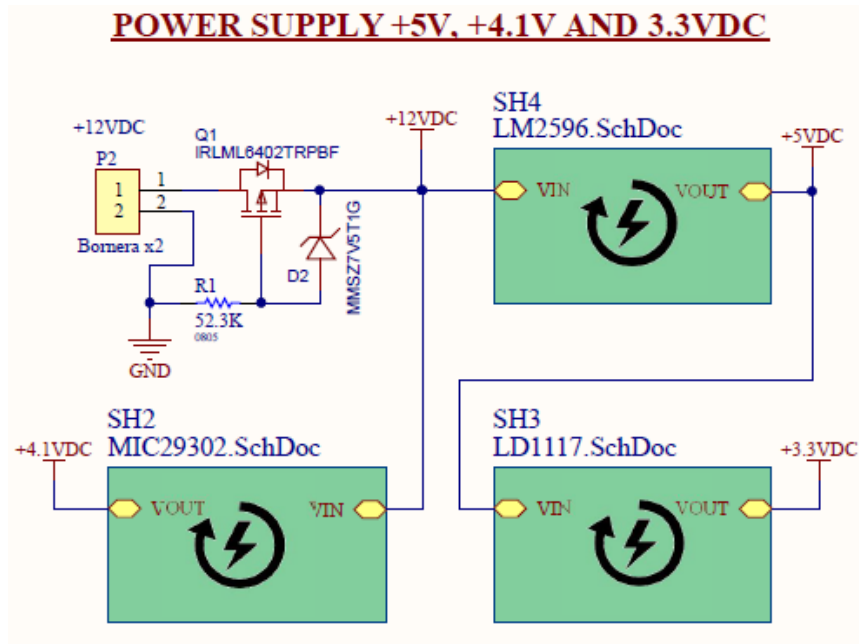


Figura 3.5 Reguladores y protección

En la siguiente figura se muestra el datasheet del módulo GSM para el envío de SMS para notificar y alertar al granjero sobre algún problema que se presente en los sensores.

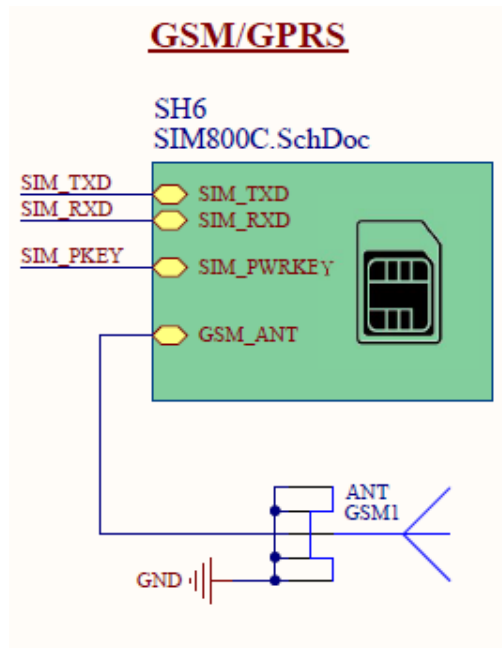


Figura 3.6 Módulo GSM para envío de SMS

En la figura 3.7 se muestra los conectores de los sensores y los respectivos actuadores, que se encargan de controlar la carga AC que puede presentarse en el prototipo.

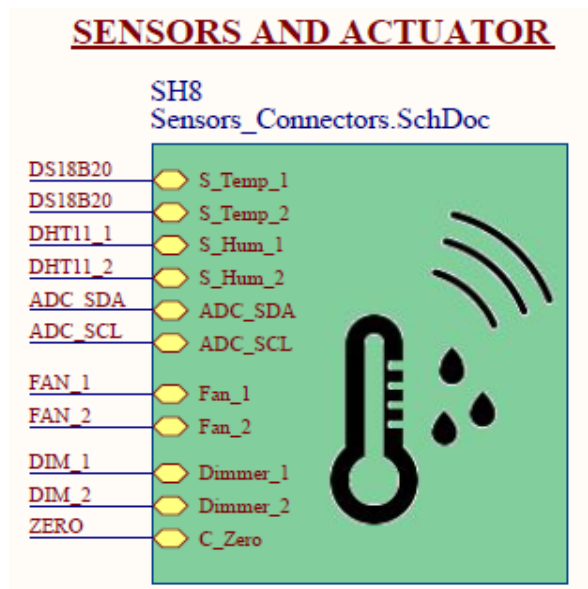


Figura 3.7 Conectores para los sensores y actuadores (relés y controladores de carga de AC)

A continuación, se muestra los diagramas de circuitos de los diferentes componentes a implementar en el prototipo IoT Avicultura:

En la figura 3.8 se muestra el bosquejo del microchip MIC29302, reguladora de tensión LDO 3.0A.

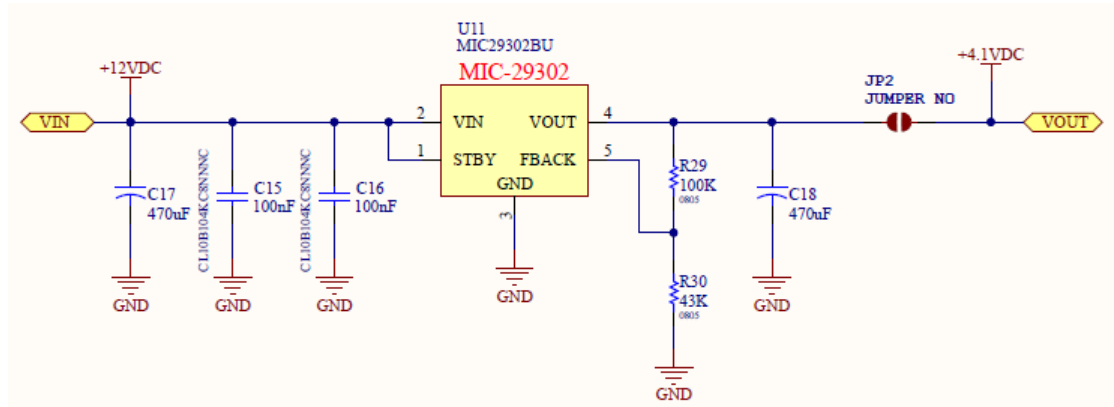


Figura 3.8 Esquemático del MIC-29302

En la figura 3.9 se muestra el bosquejo del microchip LD1117ADT33TR, regulador de tensión LDO 3.3V 1.0A.

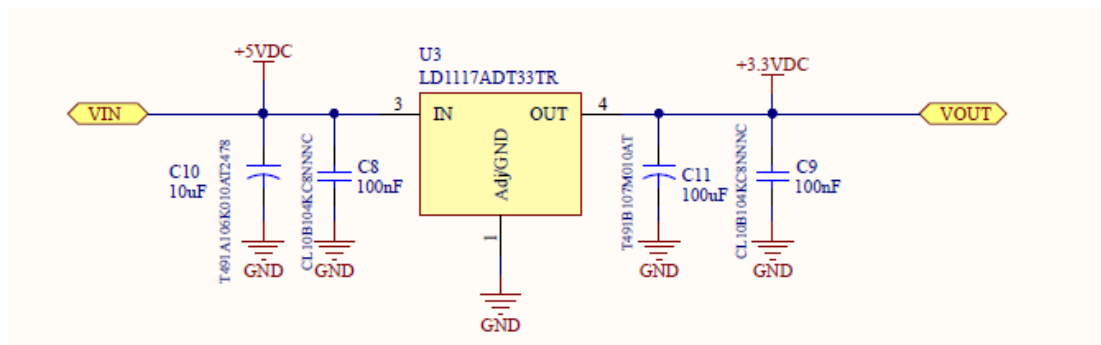


Figura 3.9 Esquemático del LD1117ADT33TR

En la siguiente figura 3.10 se muestra el bosquejo del LM2576, regulador de voltaje de conmutación 3A

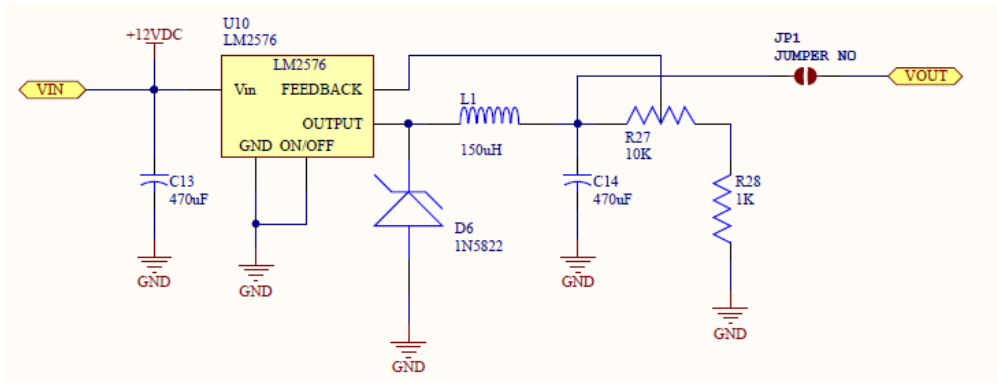


Figura 3.10 Esquemático del LM2576

En la siguiente figura 3.11 se muestra el bosquejo del ESP32, Módulo Wifi (802.11)

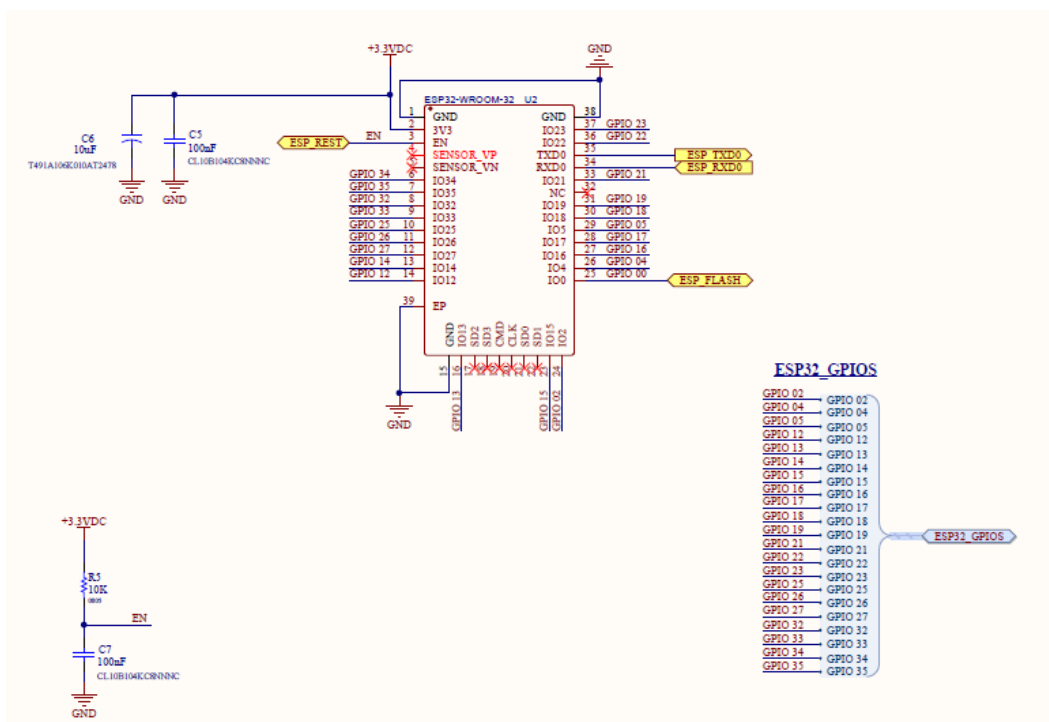


Figura 3.11 Esquemático del ESP32

En la figura 3.12 se muestra el bosquejo del SIM800C Módulos RF, GSM Module

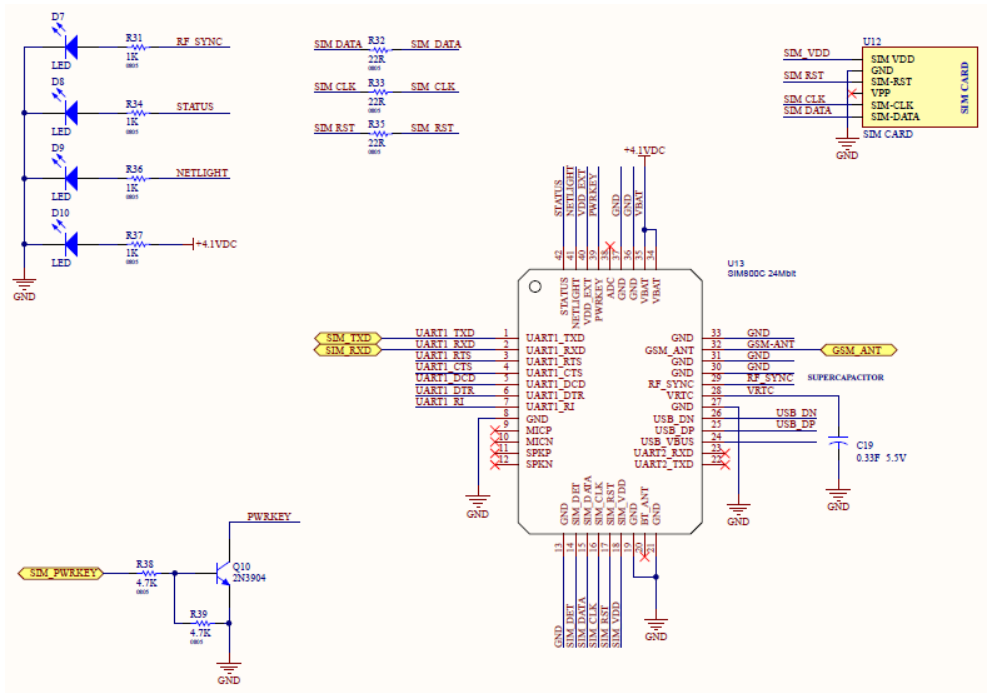


Figura 3.12 Esquemático del SIM800C

En la figura 3.13 se muestra el bosquejo del CP2102, USB-to-UART Bridge Chip

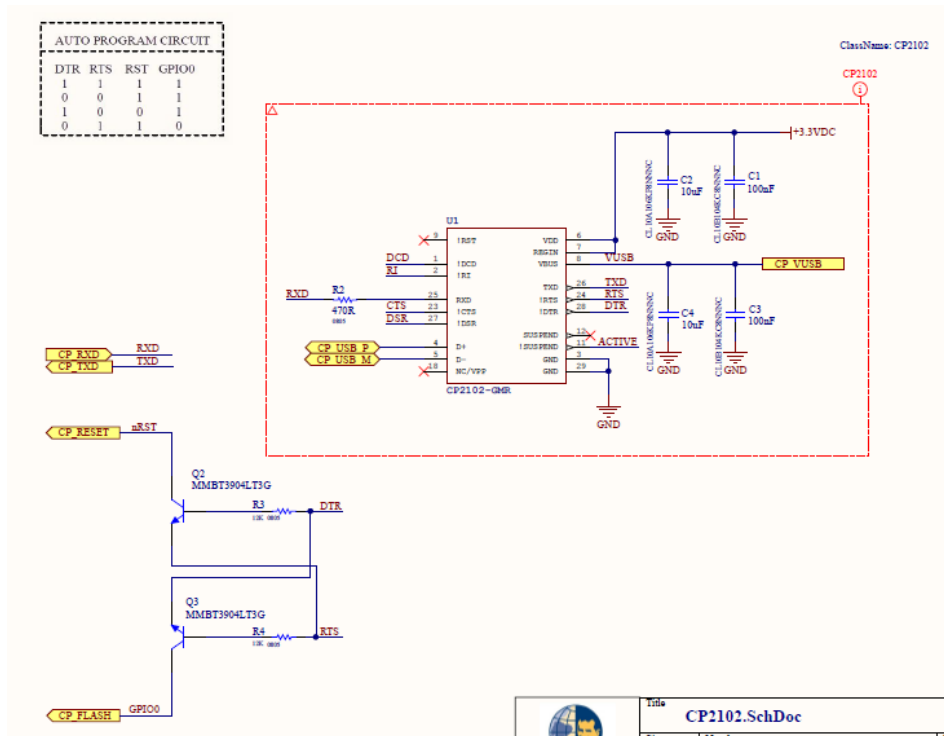


Figura 3.13 Bosquejo del CP2102

En la siguiente figura 3.14 se muestra todo el bosquejo de las conexiones de los sensores del prototipo IoT Avicultura.

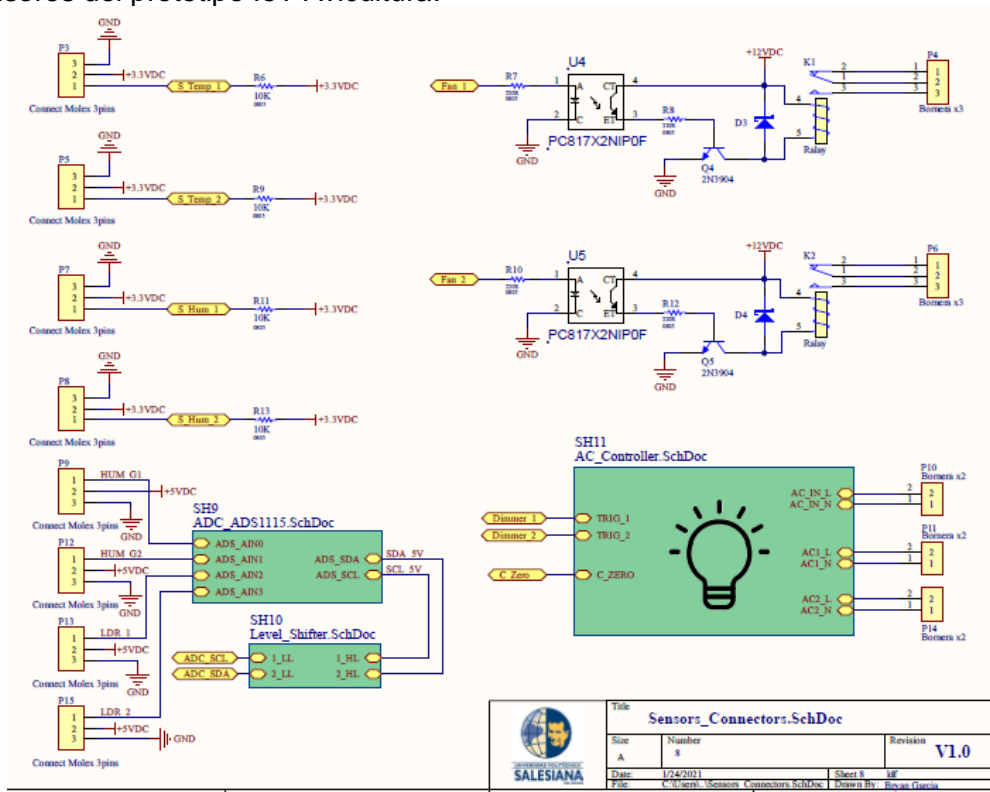


Figura 3.14 Esquemático de las conexiones de los sensores

En la figura 3.15 se aprecia el esquemático del ADS1115, Conversor analógicos a digital 16B ADC.

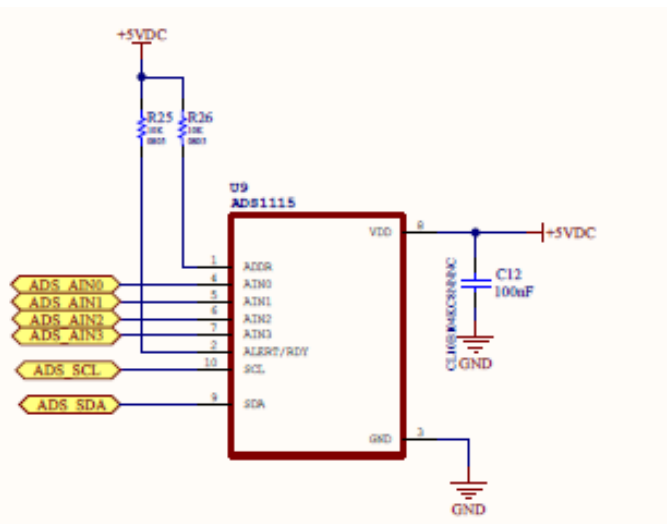


Figura 3.15 Esquemático del ADC ADS1115

En la figura 3.15 se aprecia el bosquejo del Level Shifter.

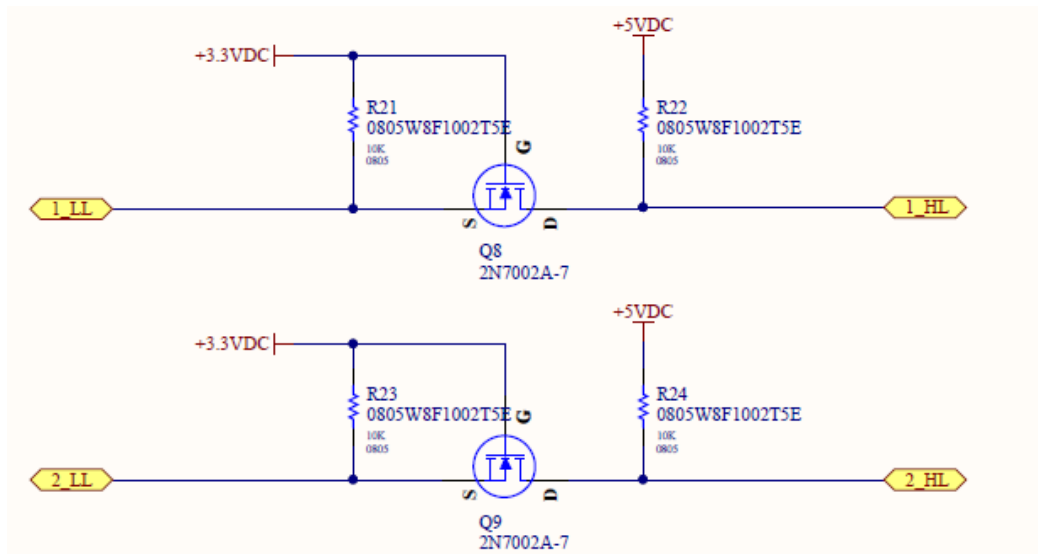


Figura 3.16 Esquemático del Level Shifter

A continuación, se aprecia el bosquejo del AC controller en la figura 3.17.

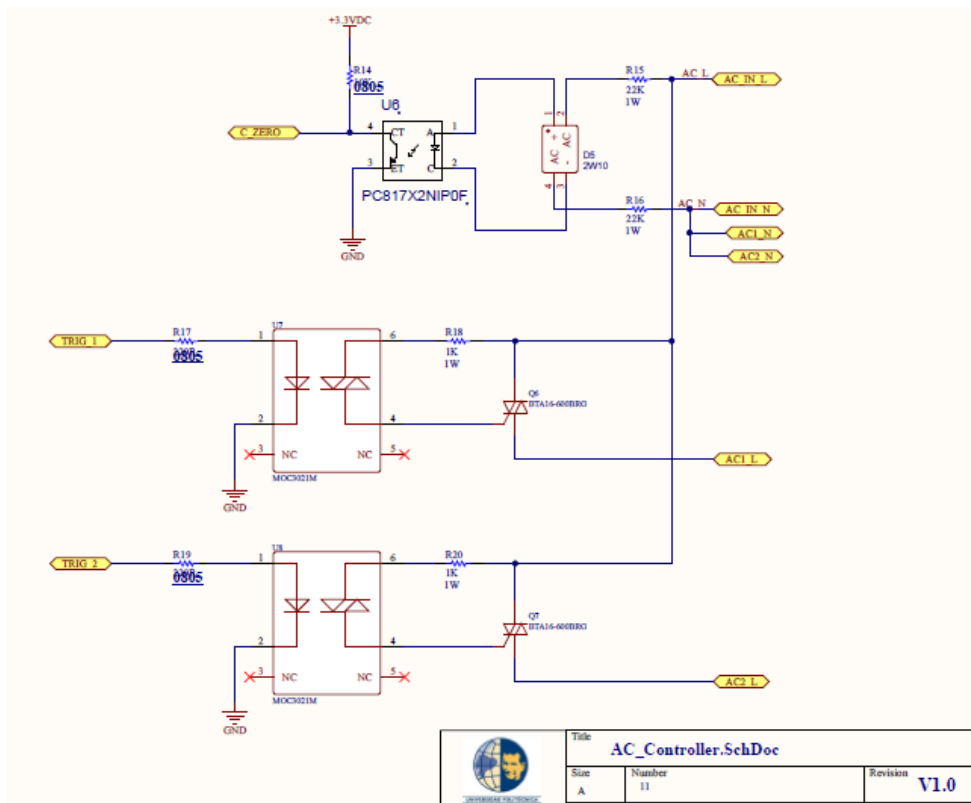


Figura 3.17 Bosquejo del AC Controller

En la figura 3.18 se observa el diseño de la placa electrónica con los diferentes componentes.

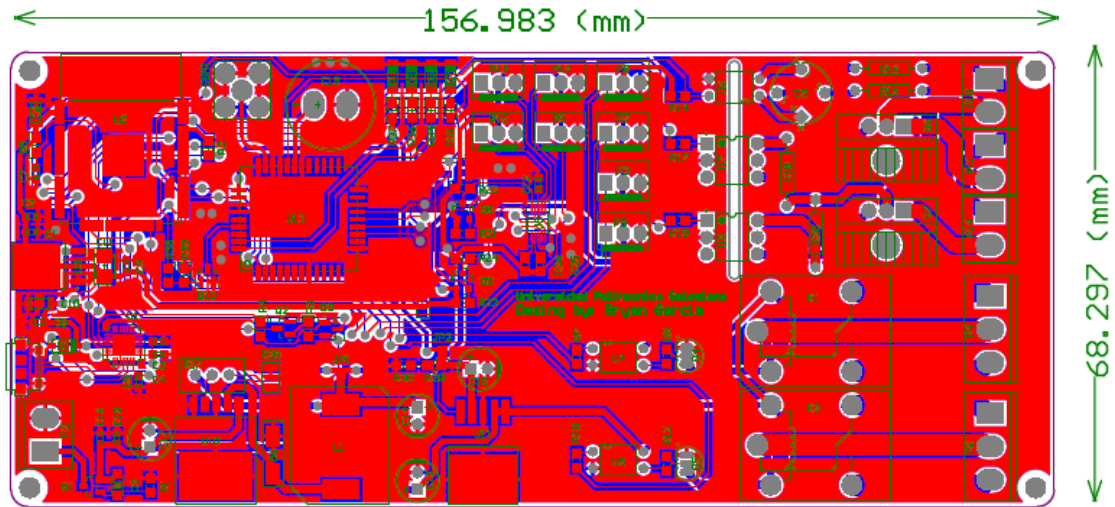


Figura 3.18 Diseño de placa electrónica

A continuación, se visualiza el diseño de la placa electrónica en 3D en la figura 3.19

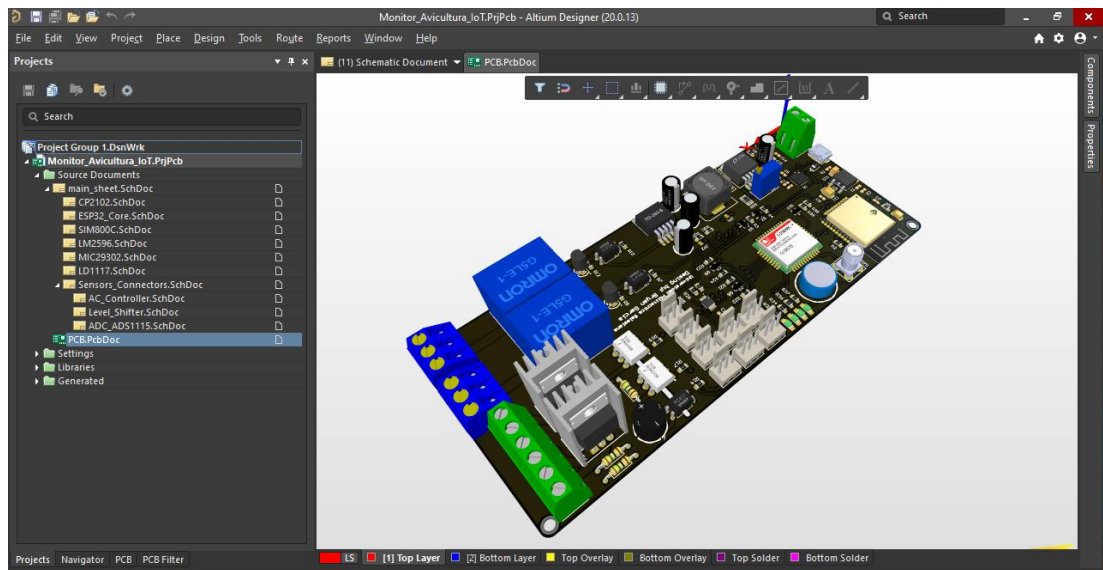


Figura 3.19 Diseño 3D placa electrónica

En la siguiente figura 3.20 se observa la parte frontal del diseño de la placa electrónica en 3D.

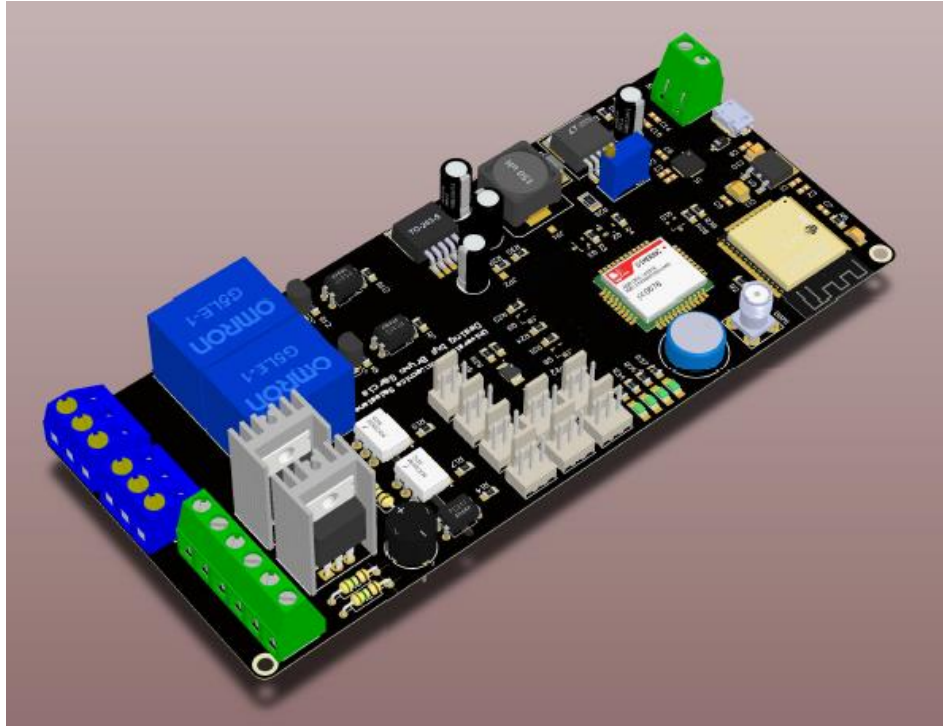


Figura 3.20 Frontal de placa electrónica

A continuación, se aprecia en la figura 3.21 el diagrama de pistas de la placa electrónica.

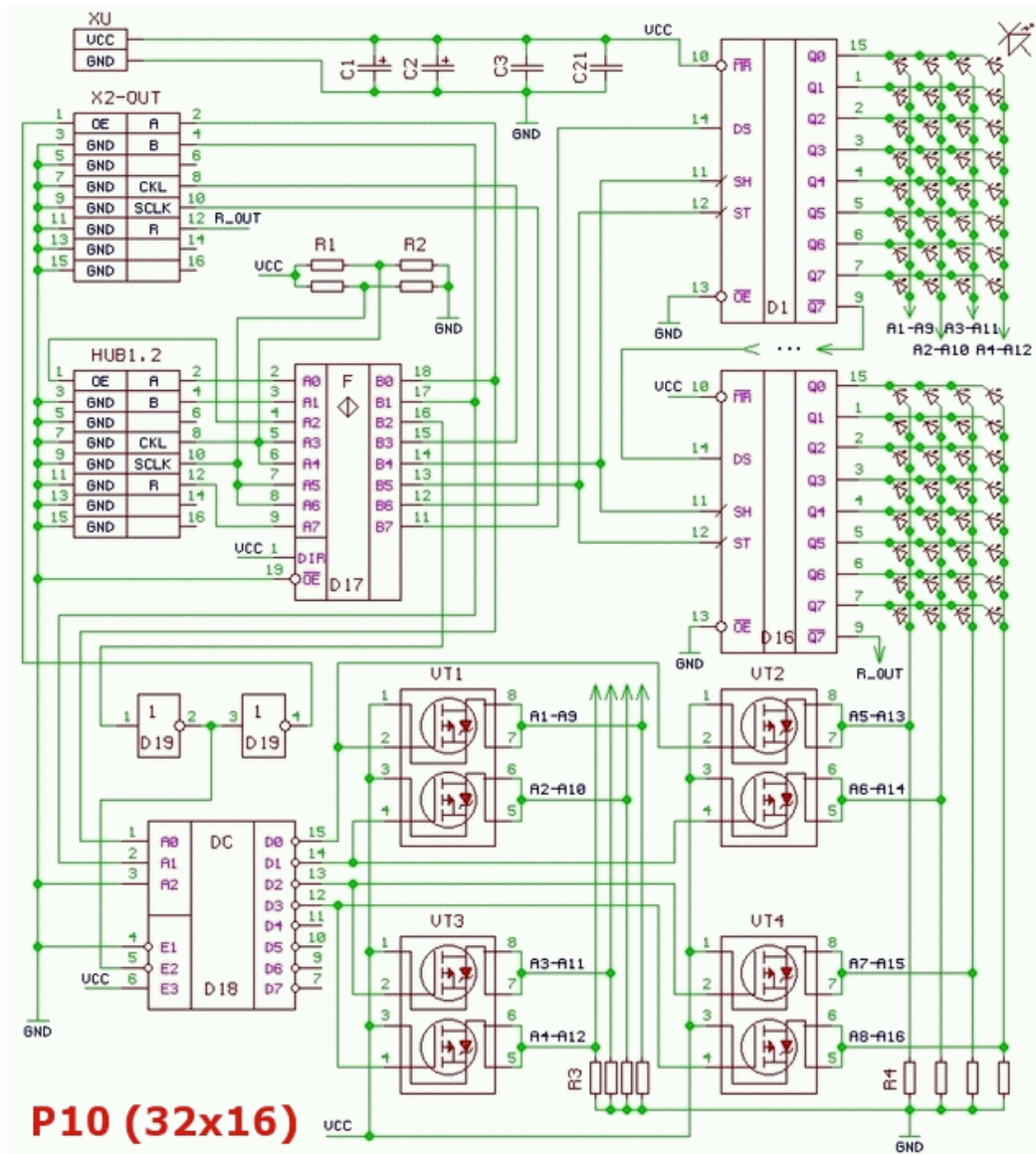


Figura 3.21 Diagrama de pistas de placa electrónica

4. Resultados

4.1. Montaje de sensores en maqueta

En esta sección se describe el montaje de los sensores utilizados en el prototipo IoT los cuales son instalados en la maqueta a escala del corral de pollitos.

En la figura 4.1 se muestra el montaje de los sensores de temperatura y humedad relativa con el que se trabaja en el prototipo IoT para avicultura.



Figura 4.1 Montaje de sensores de temperatura y humedad relativa

En la figura 4.2 se muestra la vista frontal del prototipo IoT avicultura junto con los componentes integrados.



Figura 4.2 Prototipo IoT avicultura vista frontal

En la figura 4.3 se aprecia la vista lateral del prototipo IoT avicultura junto con los componentes integrados.



Figura 4.3 Prototipo vista lateral

En la figura 4.3 se puede observar la maqueta con viruta donde se encuentran los sensores.



Figura 4.4 Maqueta con viruta

En la figura 4.5, 4.6, 4.7 y 4.8 se muestra la colocación de los sensores en la maqueta que contiene viruta del prototipo IoT avicultura.



Figura 4.5 Colocación de sensores



Figura 4.6 Colocación de sensores



Figura 4.7 Colocación de sensores



Figura 4.8 Colocación de sensores

En la figura 4.9 y 4.10 se visualiza la colocación y la conexión del ventilador que se encuentra a un costado de la maqueta del prototipo IoT Avicultura.



Figura 4.9 Colocación de ventilador



Figura 4.10 Conexión de ventilador con el prototipo

En la figura 4.11 se muestra el etiquetado de la maqueta para el prototipo IoT Avicultura.



Figura 4.11 Etiquetado de maqueta

En la figura 4.12 y 4.13 se observa la conexión de los diferentes sensores y el ventilador que se encuentran en la maqueta, que van conectados al prototipo IoT avicultura.

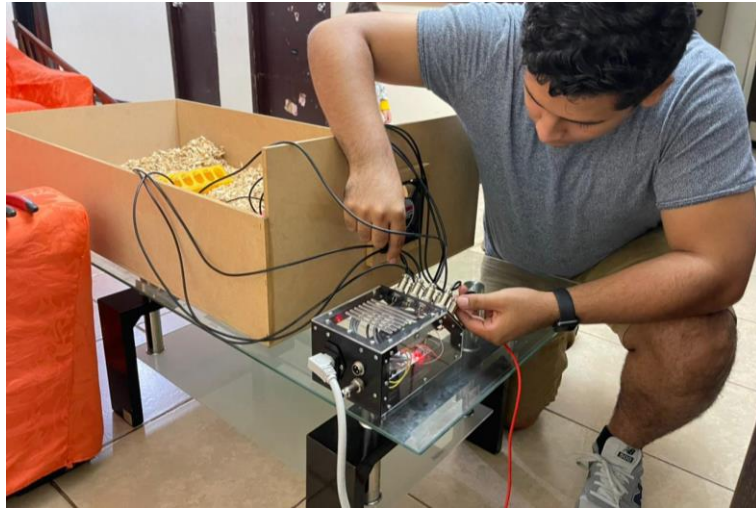


Figura 4.12 Conexión del prototipo con los sensores



Figura 4.13 Conexión de los sensores y ventilador al prototipo IoT Avicultura

4.2. Parámetros de temperatura ambiente y humedad relativa.

Los polluelos no poseen la capacidad de ajustar su temperatura corporal durante los primeros 12 a 14 días después del nacimiento. La temperatura corporal óptima debe lograrse proporcionando una temperatura atmosférica adecuada. Al momento de mudarse, la temperatura del suelo es tan importante como la temperatura del exterior, por lo que es muy destacado precalentar la casa.

El depósito debe precalentarse durante al menos 24 horas previo a la llegada de los pollitos. La temperatura y la humedad relativa (HR) deben garantizarse de acuerdo con los valores favorecidos para asegurar un ambiente confortable para el arribo de las aves. Puede ser necesario precalentar el depósito durante más de 24 horas antes del arribo de la parvada, para que la estructura interna del galpón pueda obtener efectivamente la temperatura adecuada. El tiempo necesario para calentar la casa depende del tiempo entre los ciclos de la casa y del área geográfica (las regiones donde la temperatura es bajo cero en invierno pueden demandar un tiempo de calentamiento más prolongado).

Las circunstancias ambientales que se pretenden al momento del albergue son:

- Temperatura del aire: 30°C/86°F (proporcionado a la altura del pollo en el área en la que se hallan los alimentos y el agua.
- Temperatura de la cama: de 28-30°C (82.4-86.0°F)
- HR: 60-70%

La temperatura y la humedad relativa deben monitorearse con frecuencia para asegurar la igualdad en todo el espacio de crianza. Sin embargo, el principal indicador de las circunstancias ambientales correctas es la conducta de las aves. Según el "Manual de Buenas Prácticas para la Crianza de Pollos en Gallineros", existe el siguiente termómetro en función de la edad de los pollitos, que se calcula en días.

En la tabla 4.1 se muestra los datos de temperatura adecuados durante los primeros 35 días del pollito con humedad relativa del 60%. (Bioalimentar.com 2021)

Tabla 4.1 Temperatura ambiente y humedad los primeros 35 días del pollito en el galpón.

EDAD(días)	Temperatura con 60% H.R. (° C)
0 – 2	30 – 32
3 – 6	28 – 30
7 – 9	26 – 28
10 – 12	25 – 27
13 – 15	24 – 26
16 – 18	23 – 25
19 – 21	22 – 24
22 – 25	21 – 23
26 – 30	20 – 22
31 – 35	18 – 20

Fuente: (Bioalimentar.com 2021)

4.3. Configuraciones en AWS

AWS almacena los datos en la nube por medio de Kinesis Data Analytics que es la encargada de monitorear en tiempo real de los dispositivos IoT que son implementados También proporciona servicios automáticos para la recopilación, procesamiento, análisis y visualización de las actividades y los datos que proporciona los dispositivos IoT.

En la figura 4.14 se logra apreciar la interfaz con el cual trabaja AWS para dispositivos IoT y los objetos de administración que están implementados en el prototipo IoT Avicultura.

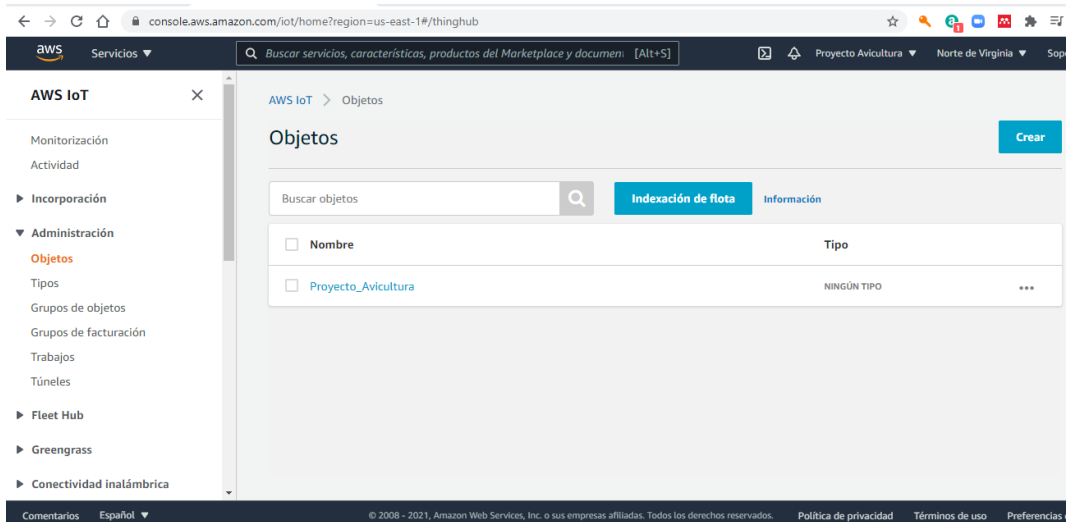


Figura 4.14 Interfaz de la consola de AWS IoT para los objetos

En la siguiente figura 4.15 se logra apreciar la interfaz de la consola de AWS para los certificados de seguridad que se implementan en el prototipo IoT avicultura.

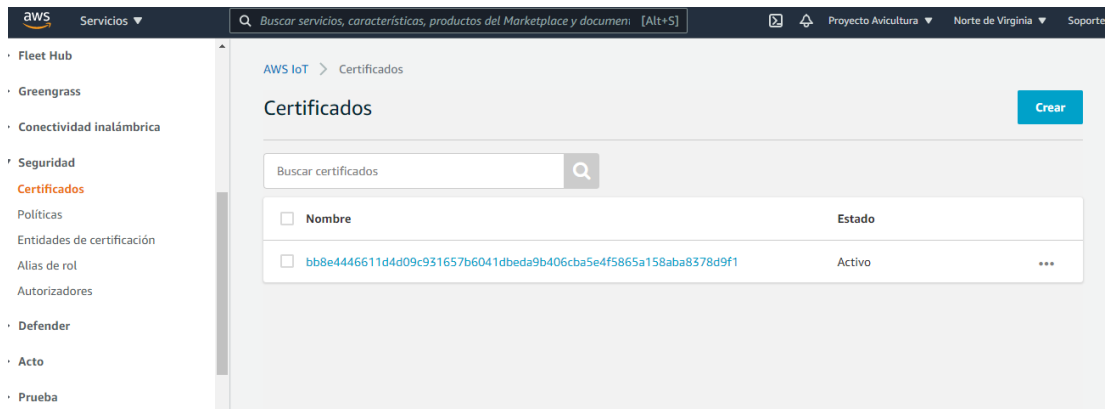


Figura 4.15 Interfaz de consola AWS para los certificados

En la figura 4.16 se observa la interfaz de la consola AWS para la políticas de seguridad de la realización del prototipo.

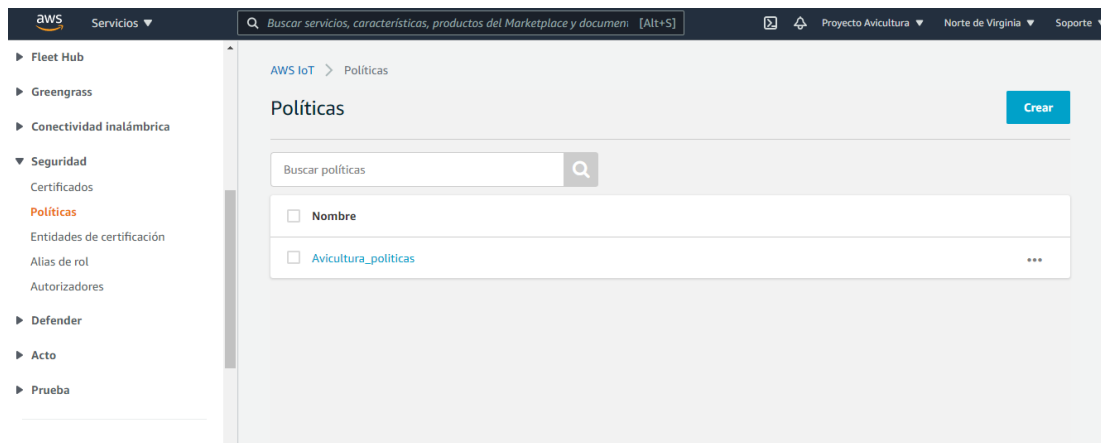


Figura 4.16 Interfaz de consola AWS para las políticas para el prototipo

4.4. Demostración del proceso de registros de los datos.

En el monitor serial se ve el ESP32 conectado al wifi programado y asignado con la dirección IP, luego lee los certificados de AWS de los SPIFFI y los muestra, también muestra los certificados en punteros formato que está pasando al cliente. Lee el ID de la Mac de ESP32 y lo muestra, pasa a iniciar la conexión MQTT con la nube de AWS conectándose. A continuación, pasa a publicar datos JSON en el tema de MQTT, se observa la temperatura y la humedad recibida en el dispositivo conectado. Todos estos datos son visualizados en AWS, en el apartado de MQTT Client (Aws.amazon.com 2021a).

4.5. Creación de reglas para AWS IoT

Las reglas que se implementan en AWS IoT permiten gestionar los datos que son enviados a través de las pruebas MQTT y almacenarlos en una plataforma de antecedentes internos que incorpora el servicio de AWS.

En la figura 4.17 se observa la regla “Data_Sensor” habilitada el cual nos permitirá mostrar los datos que son almacenamos en su correspondiente base de datos incorporada.

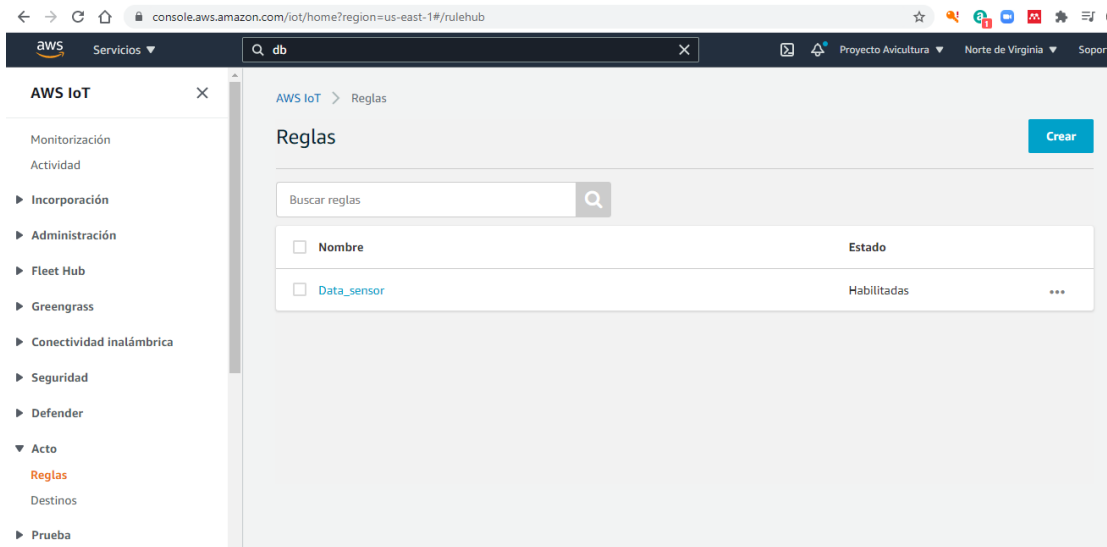


Figura 4.17 Regla de AWS IoT

En la figura 4.18 se observa la información general que contiene la regla Data_Sensor el cual es usa para mostrar y almacenar los datos que recibe desde el paquete MQTT de 'avicultura' usando SQL.

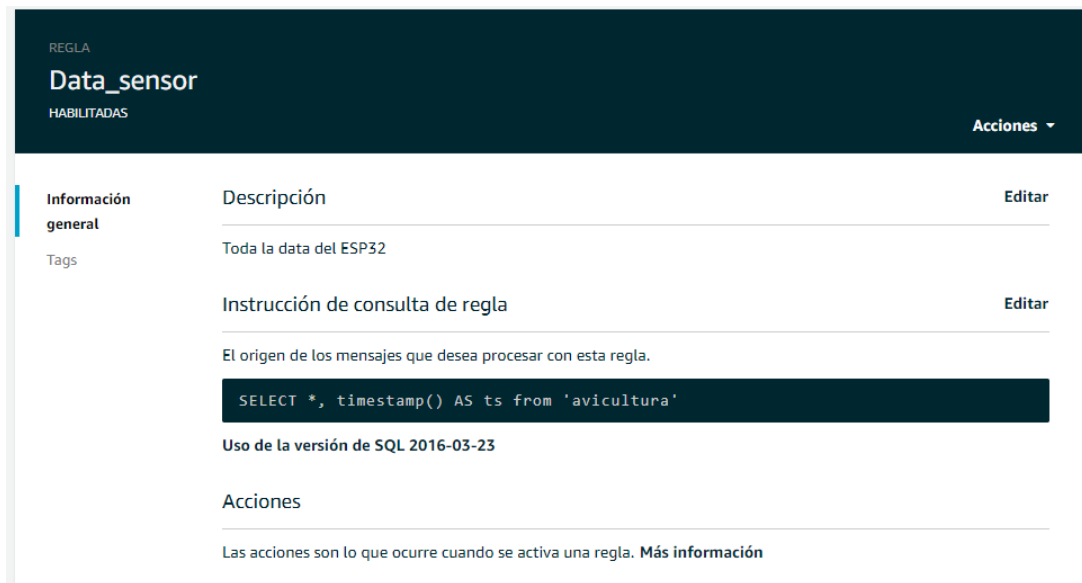



Figura 4.18 Regla Data_Sensor

En la figura 4.19 se observa las acciones y acciones de error que serán activas en la regla para el almacenamiento de datos de los sensores, para ello se implementó el dividir los mensajes en varias columnas que está incorporado en DynamoDB.

Acciones

Las acciones son lo que ocurre cuando se activa una regla. [Más información](#)

 **Dividir mensajes en varias columnas de una ...**
sensores Eliminar Editar ▶

[Añadir acción](#)

Acción de error


Si lo desea, defina la acción que se ejecutará cuando se produzca un error al procesar la regla.

[Añadir acción](#)

Figura 4.19 Acciones de la regla Data_Sensor

En la figura 4.20 se configura la acción seleccionada, para ello se crea una tabla en DynamoDBv2 con el nombre de sensores y se vincula a la regla, integrando así el almacenamiento de los datos, también se añade un rol el cual concederá los permisos en AWS IOT para ejecutar la acción configurada.

Configurar acción

 **Dividir mensajes en varias columnas de una tabla de base de datos (DynamoDBv2)**
DYNAMODBv2

La acción de DynamoDBv2 le permite escribir todo un mensaje MQTT o parte de él en una tabla de DynamoDB. Cada atributo de la carga se escribe en una columna independiente de la base de datos de DynamoDB. Los mensajes procesados por esta acción deben tener el formato JSON.

* Nombre de la tabla

↻ [Crear un nuevo recurso](#)

Elija o cree un rol para conceder a AWS IoT acceso para ejecutar esta acción.

<input type="text" value="avicultura_admin"/>	Crear un rol	Seleccionar
-----------------------------------------------	------------------------------	-----------------------------

Figura 4.20 Configuración de la Acción de la regla Data_Sensor

En la figura 4.21 se visualiza la creación de la tabla 'sensores' en DynamoDB para alojar los datos en una base de datos, el cual se obtienen de los sensores implementados en el prototipo, que llegan desde los paquetes MQTT de 'avicultura'.

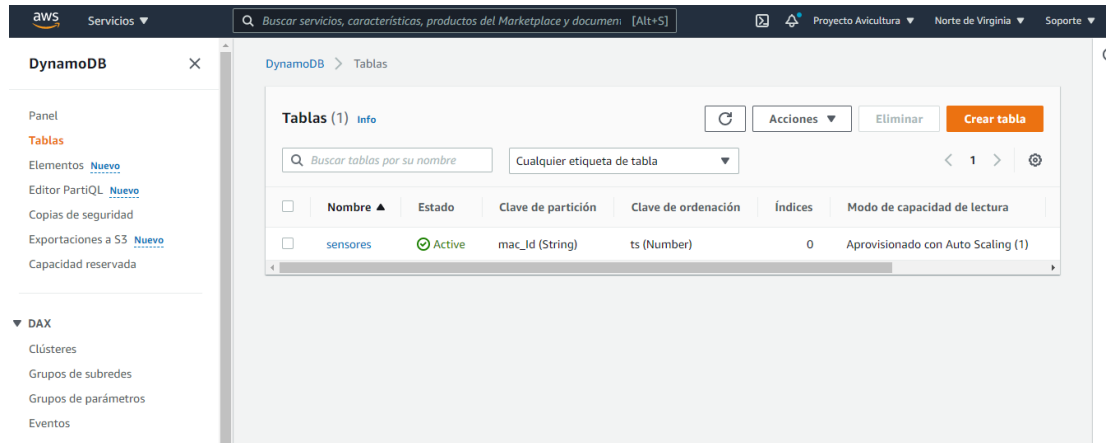


Figura 4.21 Tabla sensores en DynamoDB

En la figura 4.22 se aprecia la información general que contiene la tabla de sensores. Se puede visualizar las claves de partición, el estado de la tabla y el flujo que recibe, además de visualizar los elementos que llegan a la tabla.

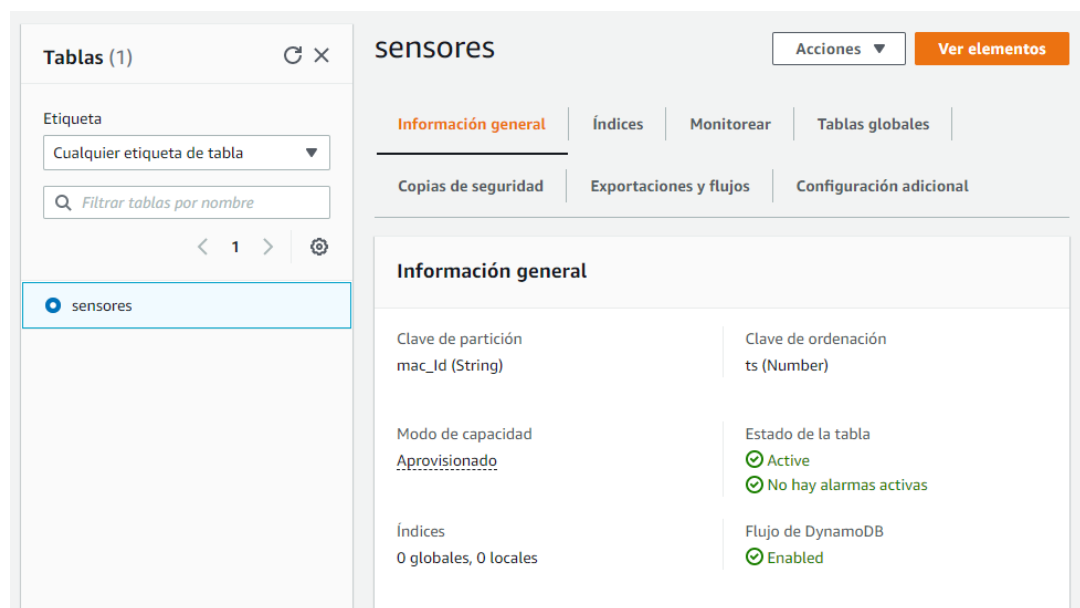


Figura 4.22 Información general de la tabla sensores

En la figura 4.23 se muestra la métrica recibida desde CloudWatch que genera la tabla de sensores durante las últimas 3 horas.

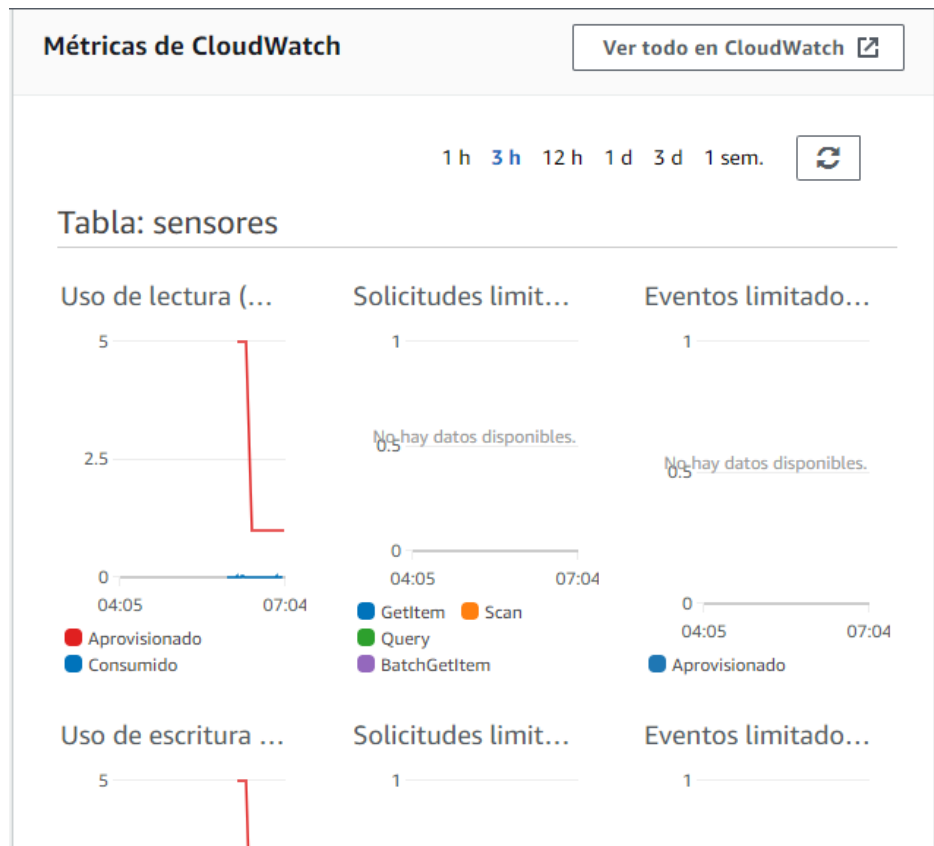


Figura 4.23 Métrica de CloudWatch de la tabla Sensores

En la figura 4.24 se visualiza la latencia que genera en la tabla de sensores de la base de datos donde esta almacenada.

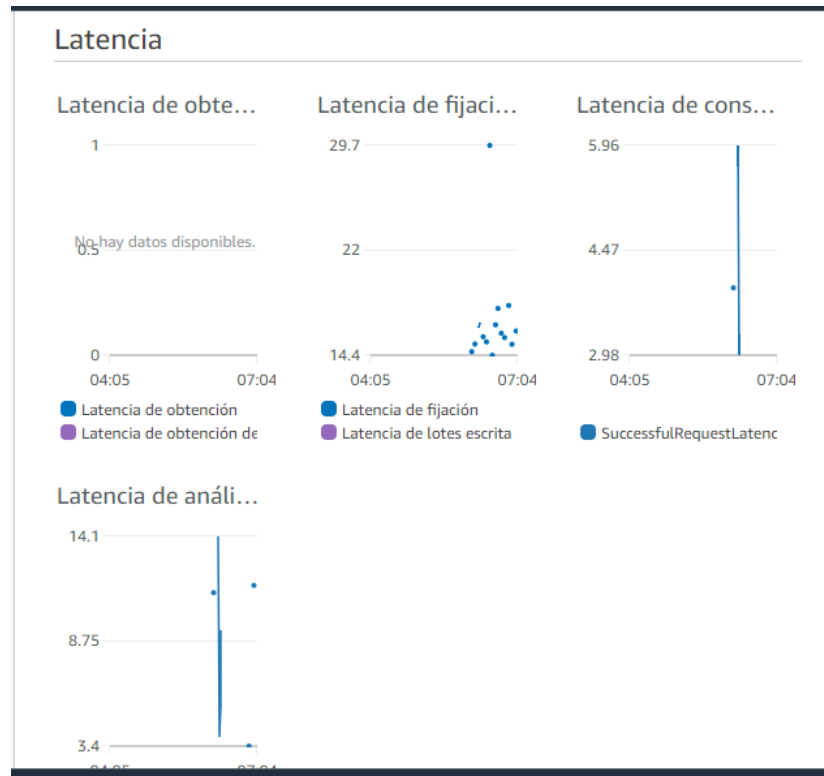


Figura 4.24 Latencia en la tabla sensores

En la figura 4.25 se visualiza las exportaciones S3 que se realiza en la tabla sensores, enviándolos a un bucket de destino para el almacenamiento de los datos.

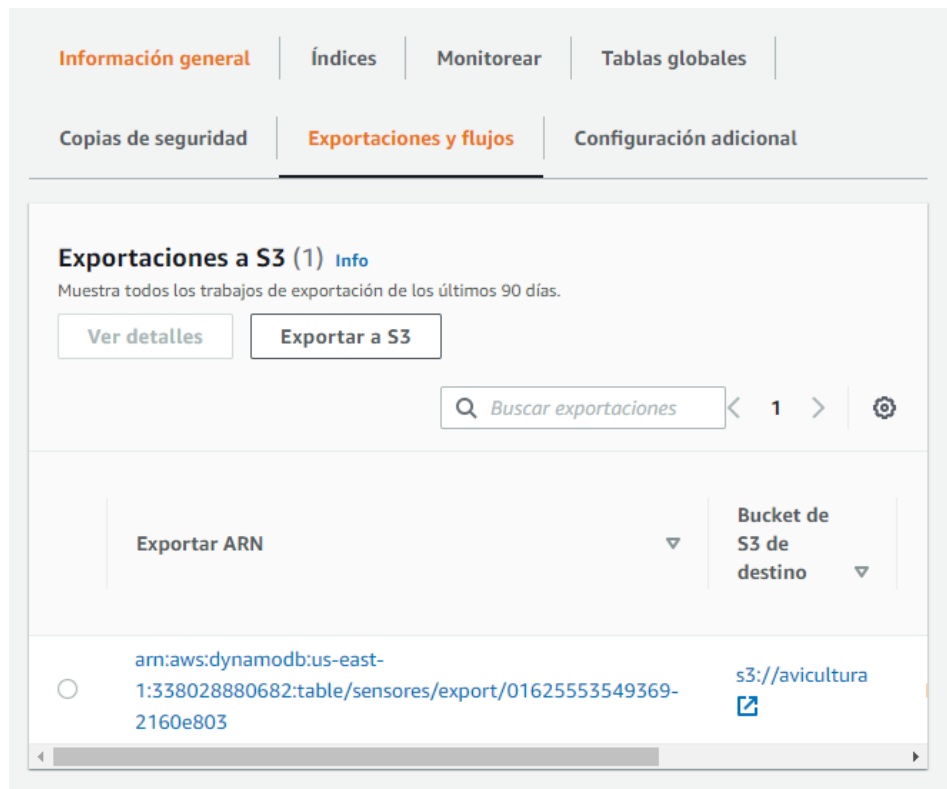


Figura 4.25 Exportaciones a S3

En la figura 4.26 y 4.27 se muestra los elementos de la tabla de sensores que llegan mediante el paquete que se envía desde el MQTT en AWS IoT. Contiene todos los valores de los sensores utilizados en el prototipo y los cuales serán almacenados en la base de datos de DynamoDB.

▼ sensores Ver detalles de la tabla

Consulta Examen

Tabla o índice
sensores

► Filtros

Ejecutar Restablecer

✔ Completado Unidades de capacidad de lectura consumidas: 0.5

Elementos devueltos (14) Acciones ▼ Crear elemento

Q Buscar elementos < 1 > ⚙️

<input type="checkbox"/>	mac_Id ▼	ts ▼	Humedad1 ▼	Humedad2 ▼	Humeda... ▼	Humeda... ▼	Luz1
<input type="checkbox"/>	7c:9e:bd:06...	162555170...	82.2	78.6	4.99	4.99	4.99

Figura 4.26 Elementos de la tabla sensores

Elementos devueltos (14) Acciones ▼ Crear elemento

Q Buscar elementos < 1 > ⚙️

<input type="checkbox"/>	mac_Id ▼	ts ▼	Humedad1 ▼	Humedad2 ▼	Humeda... ▼	Humeda... ▼	Luz1
<input type="checkbox"/>	7c:9e:bd:06...	162555170...	82.2	78.6	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555193...	82.1	78.8	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555216...	82.1	78.6	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555233...	82.1	78.8	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555256...	82	78.5	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555278...	82	78.8	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555301...	82.1	78.8	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555321...	82	78.7	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555343...	82	78.7	4.99	4.99	4.99
<input type="checkbox"/>	7c:9e:bd:06...	162555363...	82	78.7	4.99	4.99	4.99

Figura 4.27 Elementos de la tabla sensores

En la figura 4.28 se muestra las acciones que se pueden realizar dentro de la tabla 'sensores' de los elementos que son generados por los sensores que se implementan en el prototipo, logrando descargar toda la información que se muestra en la tabla en un archivo CSV.

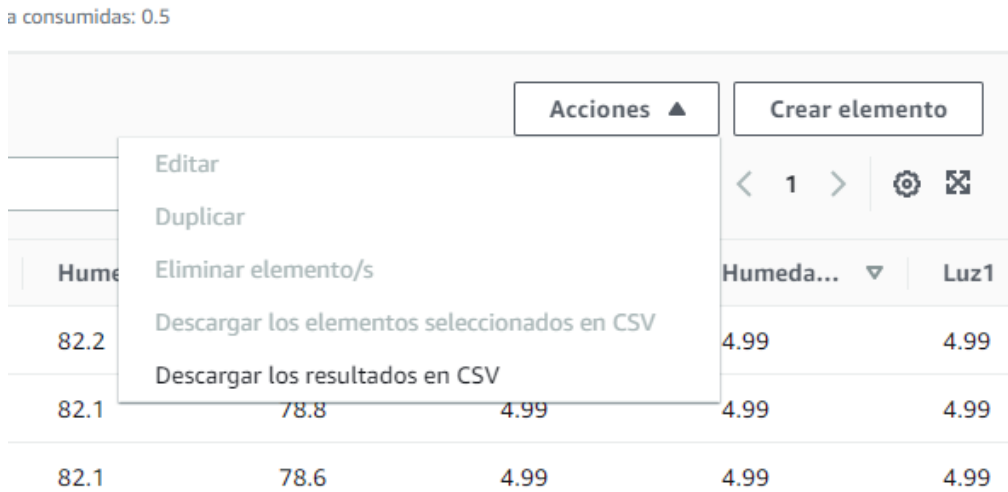


Figura 4.28 acciones de la tabla sensores

En la figura 4.29 se observa a detalle el flujo de DynamoDB de todos los registros que se almacenan dentro de la tabla y los tipos de atributos.

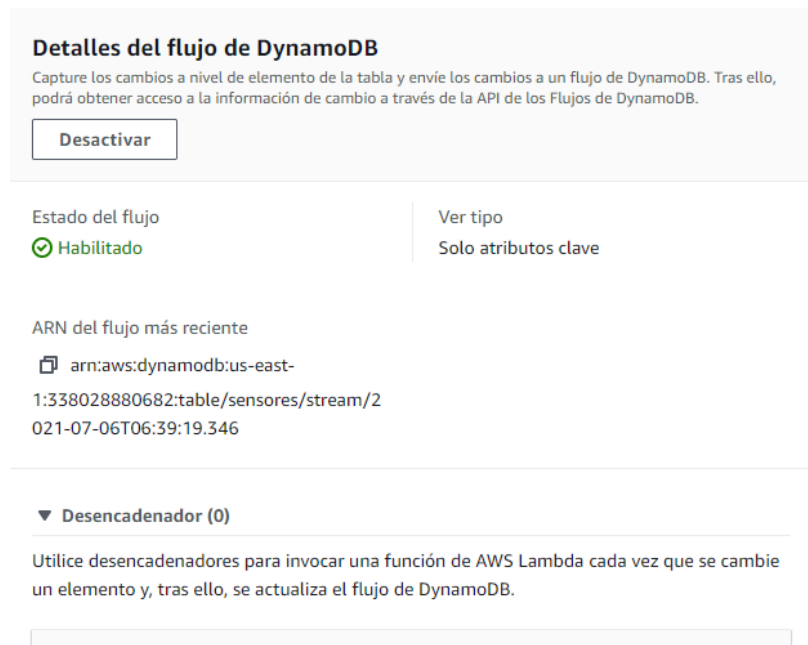


Figura 4.29 Flujo de DynamoDB

En la figura 4.30 se observa la información que recibe CloudWatch de la regla “Data_sensor” que se ha generado para el almacenamiento de los datos de los sensores que se implementan en el prototipo de Avicultura, ejecutando la consulta y visualizando los registros por segundos.

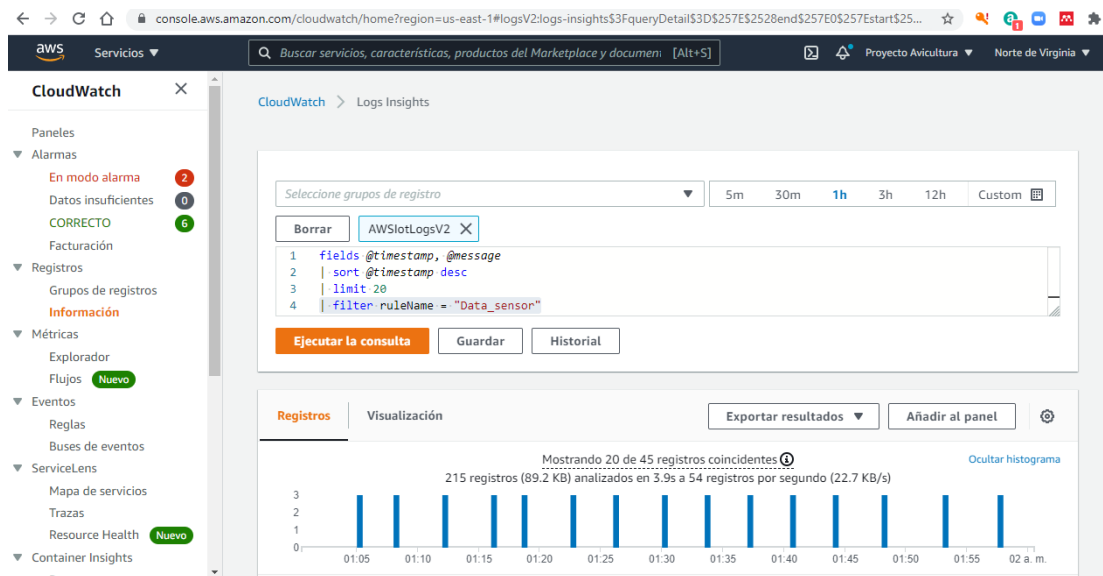


Figura 4.30 Información de CloudWatch de la regla “Data_sensor”

En la figura 4.31 se observa el usuario creado en IAM para el acceso a la información que se almacena base de datos de TimeStream, añadiendo políticas de FullAccess y acceso a aplicaciones externas para realizar la gráfica correspondiente a los datos que se mostrara en Grafana Cloud.

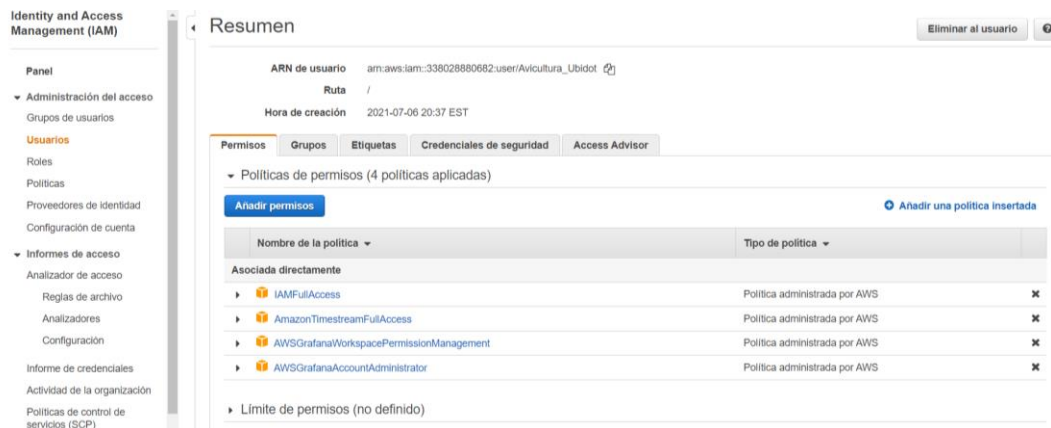


Figura 4.31 Usuario IAM

4.6. Descripción de la configuración en Grafana

Para realizar las configuraciones y la demostración de las gráficas correspondientes de los sensores, se utilizó el servicio de Grafana Cloud el cual permite mostrar los datos que nos provee los sensores que están conectados en AWS IoT, y luego son almacenamos en AWS TimeStream, para el análisis de datos dentro de su base de datos.

A continuación, se mostrará la configuración que se realizó a cada uno de los sensores incorporados en el prototipo, mediante la base de datos de AWS TimeStream. En las figuras del 4.32 al 4.41 se muestra las configuraciones realizadas en Grafana y en los Querys para la obtención de la data del servidor de TimeStream.



Figura 4.32 Configuración de Temperatura1 en Grafana



Figura 4.33 Configuración de Temperatura2 en Grafana

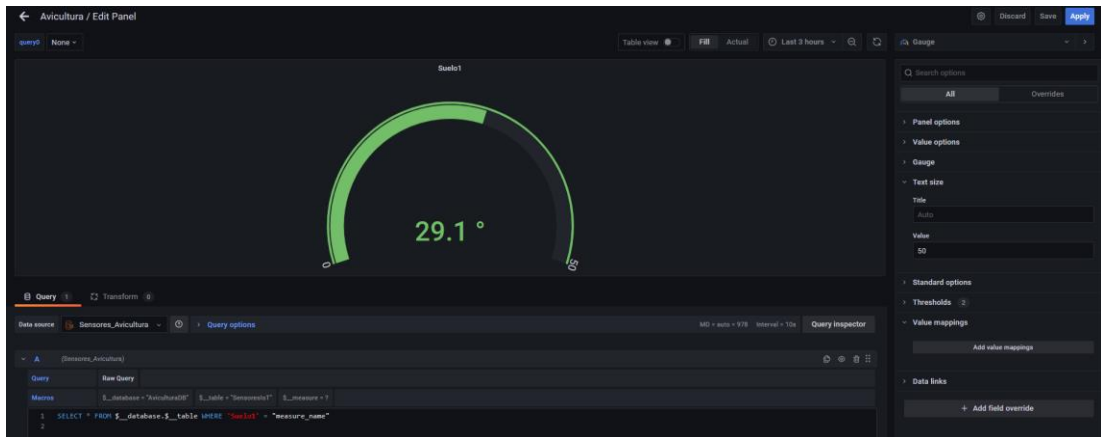


Figura 4.34 Configuración de Suelo1 en Grafana

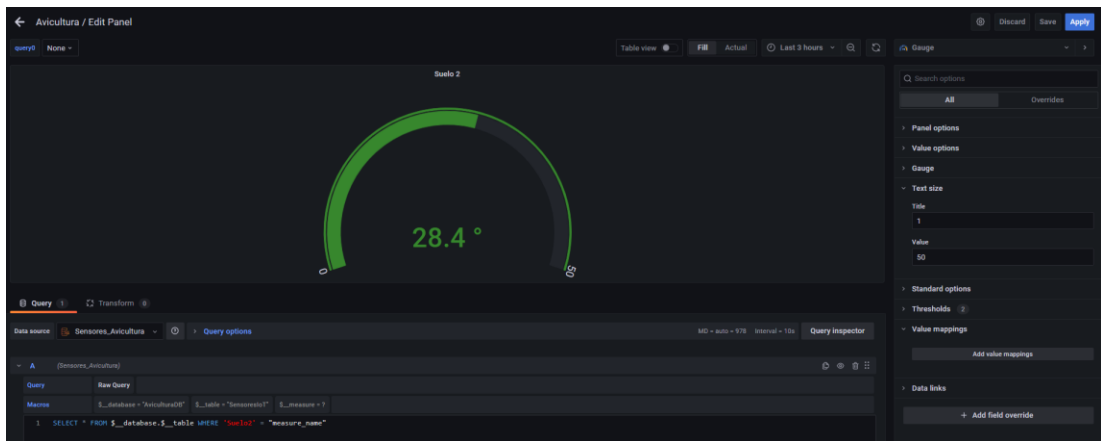


Figura 4.35 Configuración de Suelo2 en Grafana

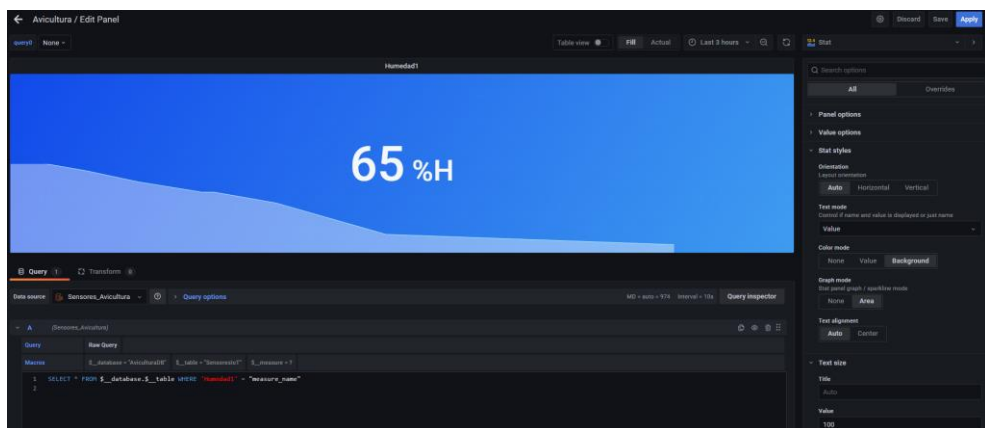


Figura 4.36 Configuración de Humedad1 en Grafana

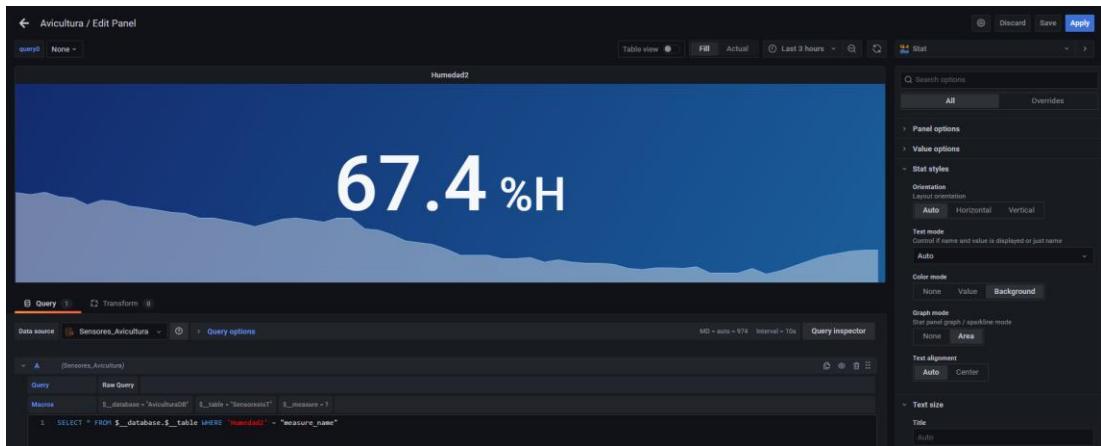


Figura 4.37 Configuración de Humedad2 en Grafana

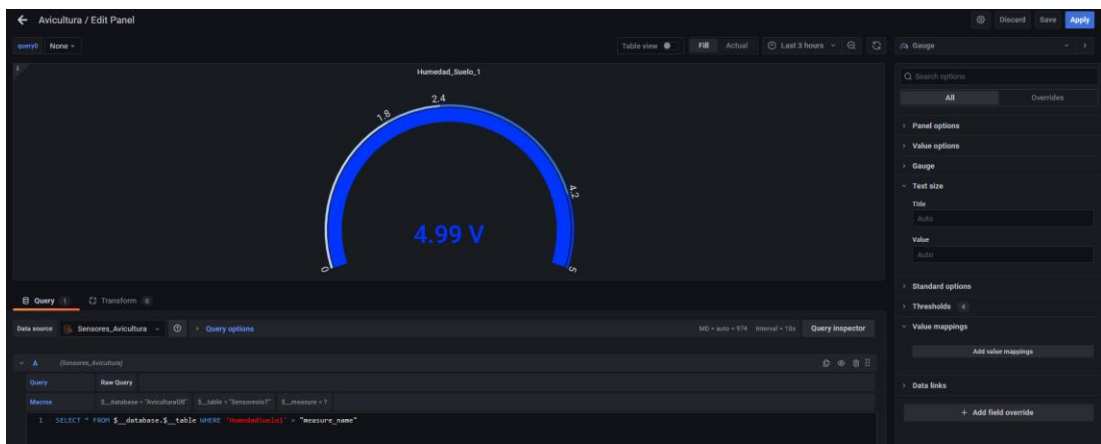


Figura 4.38 Configuración de HumedadSuelo1 en Grafana



Figura 4.39 Configuración de HumedadSuelo2 en Grafana



Figura 4.40 Configuración de Luz1 en Grafana



Figura 4.41 Configuración de Luz2 en Grafana

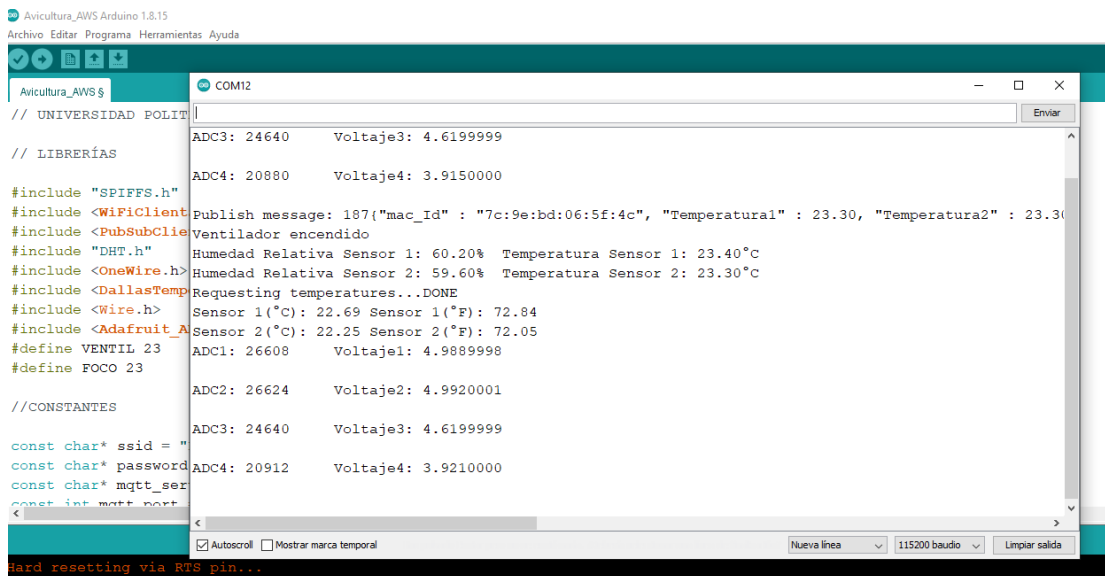
5. Análisis de resultados

5.1. Monitor de sensores desde IDE de Arduino

Para la monitorización de los sensores se utiliza el IDE de Arduino, donde se procede a escribir los códigos de programación para que los diferentes sensores que están implementados funciones y poder así obtener resultados con cada uno de ellos como se muestra en la figura 5.1, donde se visualiza el voltaje, la temperatura, si el ventilador esta activado, la humedad relativa de los sensores e intensidad lumínica.

Además, se debe agregar líneas de comando para vincular el enlace con AWS IOT y respaldar los datos en la nube. En la configuración de void setup () se añade el archivo SPIFFS para la lectura de archivos de AWS.

El código configurado en el ESP32 se detalla en el anexo de este libro.



```
Avicultura_AWS Arduino 1.8.15
Archivo Editar Programa Herramientas Ayuda

Avicultura_AWS $
// UNIVERSIDAD POLIT
// LIBRERÍAS
#include "SPIFFS.h"
#include <WiFiClient
#include <PubSubClie
#include "DHT.h"
#include <OneWire.h>
#include <DallasTemp
#include <Wire.h>
#include <Adafruit_A
#define VENTIL 23
#define FOCO 23

//CONSTANTES
const char* ssid = "
const char* password
const char* mqtt_ser
const int mqtt_port

ADC3: 24640 Voltaje3: 4.6199999
ADC4: 20880 Voltaje4: 3.9150000
Publish message: 187{"mac_Id" : "7c:9e:bd:06:5f:4c", "Temperatura1" : 23.30, "Temperatura2" : 23.30}
Ventilador encendido
Humedad Relativa Sensor 1: 60.20% Temperatura Sensor 1: 23.40°C
Humedad Relativa Sensor 2: 59.60% Temperatura Sensor 2: 23.30°C
Requesting temperatures...DONE
Sensor 1(°C): 22.69 Sensor 1(°F): 72.84
Sensor 2(°C): 22.25 Sensor 2(°F): 72.05
ADC1: 26608 Voltaje1: 4.9889998
ADC2: 26624 Voltaje2: 4.9920001
ADC3: 24640 Voltaje3: 4.6199999
ADC4: 20912 Voltaje4: 3.9210000

Autoscroll [x] Mostrar marca temporal [x] Nueva línea [v] 115200 baudio [v] Limpiar salida [v]
Hard resetting via RTS pin...
```

Figura 5.1 Ejecución del monitoreo de Sensores desde el IDE en Arduino

Una vez leída las cadenas que se le envía, se convertirán en punteros ya que el espCliente no acepta cadenas, se imprimen los punteros convertidos para compararlos con el original y verificar los resultados. Una vez estos punteros son convertidos a ESPCliente, se inicia el servidor MQTT END Point y el puerto. Siendo posible leer la dirección Mac del módulo ESP32.

Una vez que se finaliza el conteo de cadena se leen respectivamente las lecturas de temperatura y humedad de los sensores y mostrándolos en el dispositivo vinculado.

Se comprueba la conexión al cliente, en el caso de que no esté conectado se llama a la función de reconexión para establecer la conexión.

Por último, se procede a convertir las variables que ingresan de los sensores a formato de cadena para volver a convertirlos al formato JSON, para que el ESP32 muestre por medio de parpadeos que la información llegó con éxito al AWS.

Una vez ingresada toda la información en el IDE de Arduino se cargará a la placa de ESP32 para el funcionamiento de los sensores.

5.2. Pruebas de envío de mensajes GSM

El envío de mensajes de alerta SMS en la red GSM se realiza mediante la configuración de los umbrales definidos en el código del ESP32.

En la figura 5.2 se observa que los SMS llegan al número configurado previamente en el código del ESP32, para las pruebas y análisis de resultados se ha configurado los umbrales de acuerdo con las recomendaciones de los manuales prácticos de la crianza de pollos en galpones.

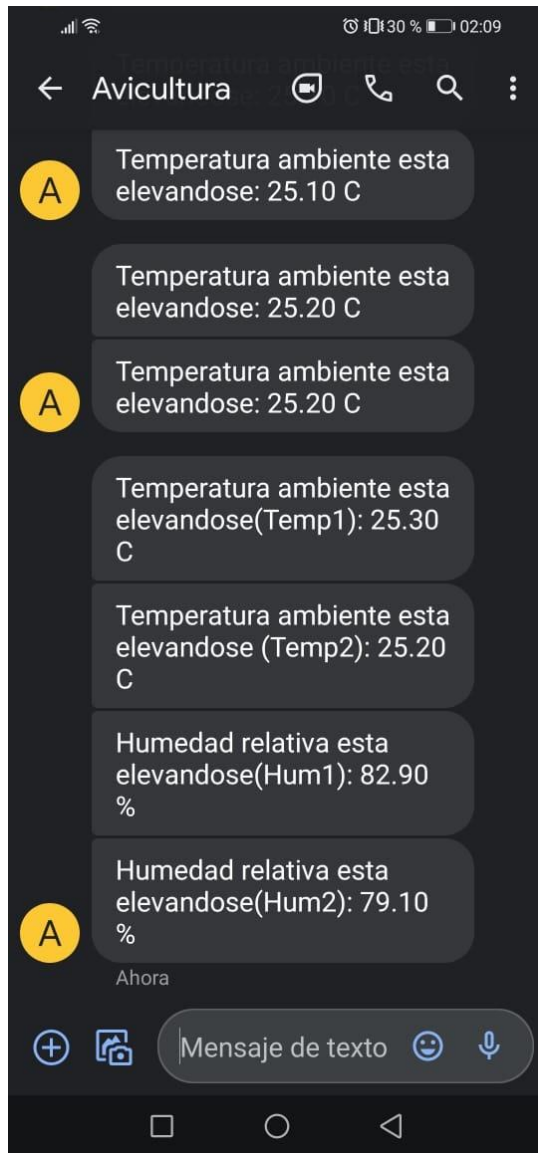


Figura 5.2 SMS de alerta recibidos

5.3. Monitor de sensores desde AWS

Para realizar el monitoreo de sensores en AWS, se debe crear un objeto en la administración en AWS IOT, una vez creado se generará un certificado y claves. Una vez que se genera, se procede a realizar las políticas que se definirán en AWS para los dispositivos IoT.

En la figura 5.3 se muestra la interfaz del monitoreo de los sensores desde la AWS de Amazon.

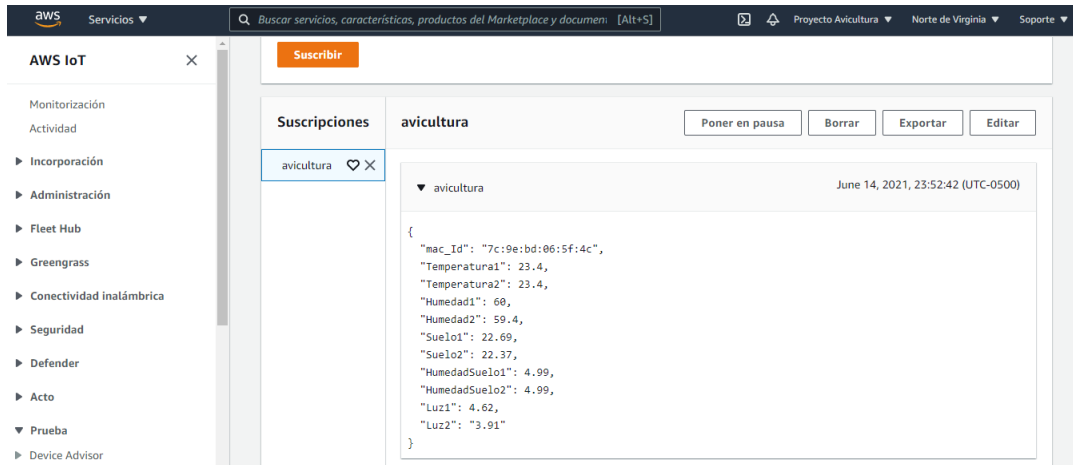


Figura 5.3 Monitoreo de los sensores desde AWS IoT

5.4. Visualización de datos en Grafana Cloud

En la figura 5.4 se muestra el acceso a través de Grafana a la base de datos de Amazon TimeStream para visualizar la correspondiente tabla y sus datos generados

por los sensores del prototipo de Avicultura.

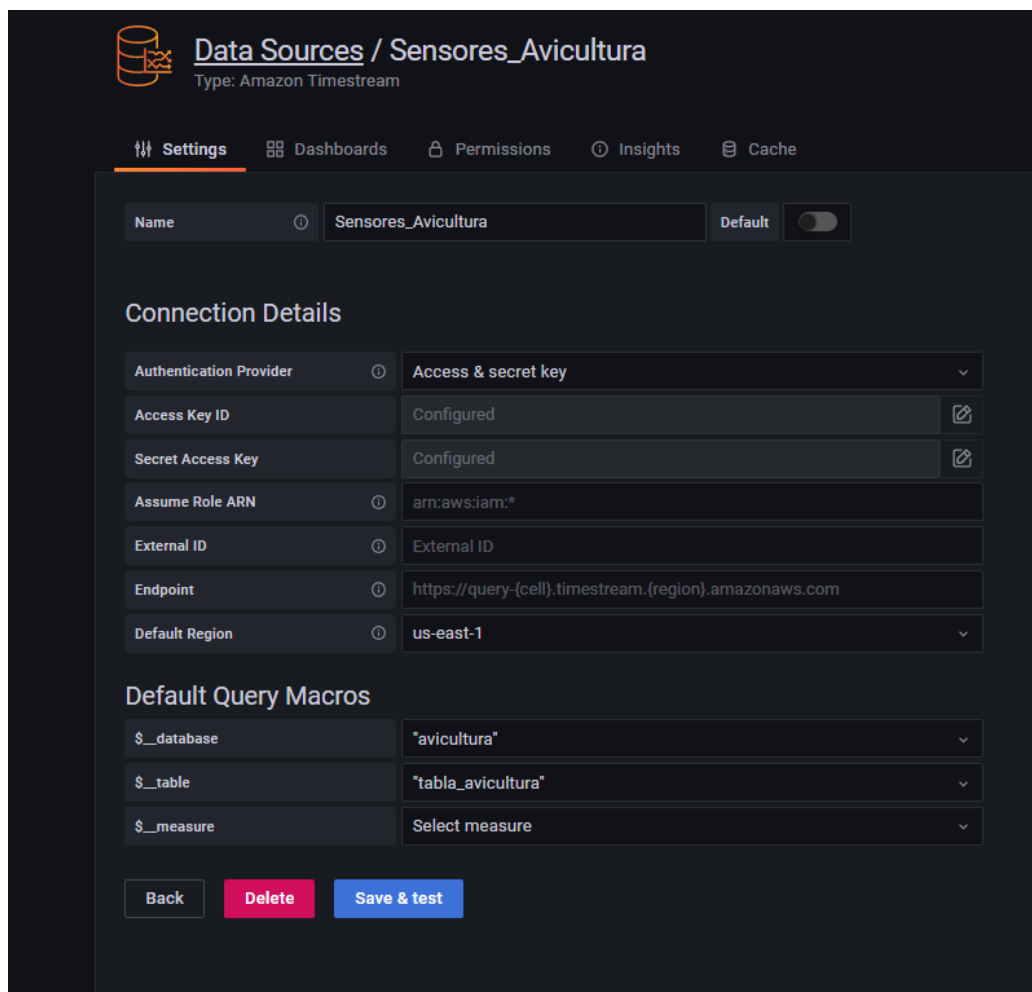


Figura 5.4 Acceso Grafana desde TimeStream

En la figura 5.5 se muestra los datos de los sensores del prototipo, que se obtienen mediante los registros almacenados en la base de datos de TimeStream en Amazon.



Figura 5.5 Registros de los datos mediante Grafana

En las siguientes figuras se detallan las gráficas de cada sensores.



Figura 5.6 Monitorización de Temperatura ambiente 1 y 2.

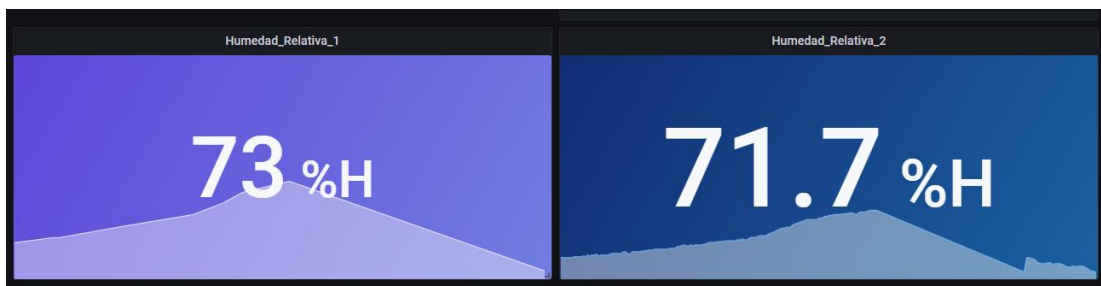


Figura 5.7 Monitorización de humedad relativa 1 y 2.

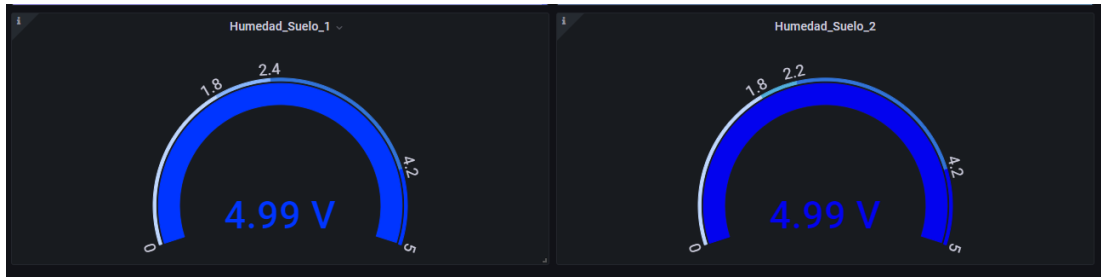


Figura 5.8 Monitorización de humedad de suelo 1 y 2.

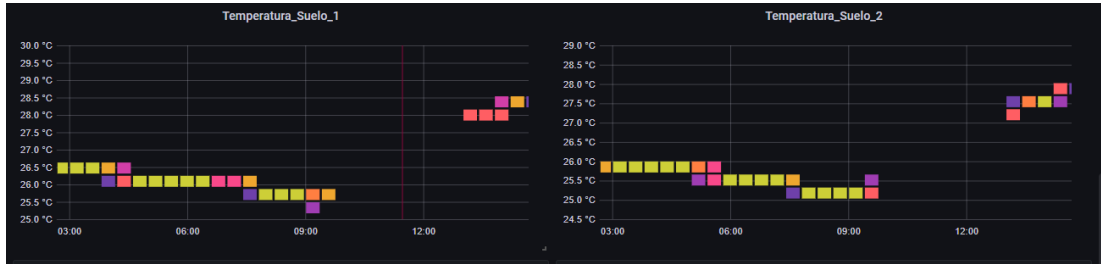


Figura 5.9 Monitorización de temperatura de suelo 1 y 2.



Figura 5.10 Monitorización de intensidad de luz 1 y 2.

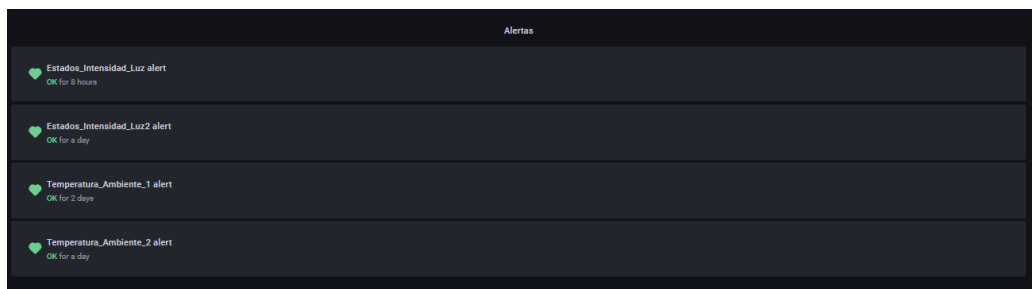


Figura 5.11 Alarmas de sensores.

6. Conclusiones

Como conclusión del trabajo de investigación se tiene los siguientes puntos importantes:

- Se diseñó e implementó un prototipo IoT para el monitoreo de parámetros ambientales aplicados a la avicultura para la crianza de pollos de granja utilizando ESP32 y servidores web en la nube con AWS. Los materiales utilizados en la construcción y ensamblaje del prototipo fueron de bajo costo.
- Se diseñó una red de sensores de temperatura y humedad ambiental, humedad de suelo e intensidad lumínica para dispositivo IoT de monitorización de parámetros ambientales para la crianza de pollos de granja. Para el prototipo se utilizó dos sensores de cada uno ubicados de manera estratégica en el galpón maqueta. Para futuros desarrollos en galpones reales estos sensores pueden incrementar y el diseño se debe expandir.
- Se configuró servicios en la nube para el almacenamiento y monitorización de la información de los sensores con Amazon Web Services. Para esto se utilizó AWS IoT Core para obtener la data de los sensores.
- Se diseñó un sistema de alerta SMS para la monitorización de umbrales de los parámetros de temperatura, humedad, humedad de suelo e intensidad de luz para la crianza de pollos de granja. Esta opción es importante para zonas donde no hay cobertura de internet, se mantendrá solo el monitoreo vía SMS.
- Se realizó un manual de práctica del prototipo IoT de monitorización ambiental para la crianza de pollos de granja. El manual ayuda al estudiante de ingeniería en telecomunicaciones y electrónica a utilizar correctamente el prototipo IoT y realizar futuras mejoras.

7. Recomendaciones

Las recomendaciones más importantes de este proyecto de titulación son las siguientes:


- Para la implementación del prototipo IoT de avicultura en galpones grandes, se debe extender el cableado de cada sensor, ya que lo estructurado en este proyecto es para maqueta.
- Para el envío de data mediante internet se requiere de una ancho de banda mínimo de 2 Mbps para el envío de la data a AWS IoT Core.
- Para extender la cobertura de la red WIFI del ESP32 se puede adquirir un repetidor WIFI para que el ESP32 tenga buena conectividad de la red.
- En caso de que en el galpón de pollos no haya cobertura de internet se sugiere revisar que exista cobertura GSM para el envío de mensajes SMS de alerta.
- Se debe tener una tarjeta de crédito o débito para costear los servicios de AWS IoT ya que la cuenta es gratuita hasta cierto límite de uso, al sobrepasar este límite de uso mensual se realizará débito bancario de la tarjeta, para este proyecto y durante las pruebas se ha invertido aproximadamente \$25 mensual.
- AWS consta de un servicio de soporte técnico premium para desarrolladores con un costo de \$29 mensuales, se puede adquirir este servicio en caso de requerir ayuda o soporte técnico especializado para los servicios de AWS.
- El prototipo IoT puede estar a temperatura ambiente sin embargo se recomienda colocarlo en una zona fresca, sin humedad y con protección a los rayos solares.

Bibliografía

- Amazon Web Services. 2020. "¿Qué Es AWS?" Retrieved November 3, 2020 (https://aws.amazon.com/es/what-is-aws/?nc2=h_ql_le_int).
- Arturo, Carlos, Monje Álvarez, Universidad Surcolombiana, Facultad De, Ciencias Sociales, Y. Humanas, Programa De Comunicación Social, Y. Periodismo, and Carlos Arturo Monje Álvarez. 2011. *METODOLOGÍA DE LA INVESTIGACIÓN CUANTITATIVA Y CUALITATIVA Guía Didáctica*.
- avicultura.info. 2021. "Información Avicultura Región Andina - Artículos, Noticias y Mucho Más." Retrieved February 6, 2021 (<https://avicultura.info/paises/region-andina/>).
- Avicultura.info. 2018. "¿Cómo Se Vincula Una Granja Avícola y La Internet de Las Cosas? - AviNews, La Revista Global de Avicultura." Retrieved February 6, 2021 (<https://avicultura.info/como-se-vincula-una-granja-avicola-y-la-internet-de-las-cosas/>).
- Avicultura.info. 2020. "Mortalidad Avícola, Una Herramienta Útil Para Las Explotaciones." Retrieved October 22, 2020 (<https://avicultura.info/caracterizacion-de-la-mortalidad/>).
- aws.amazon.com. 2021. "Amazon Timestream – Base de Datos de Series Temporales – Amazon Web Services." Retrieved July 11, 2021 (<https://aws.amazon.com/es/timestream/>).
- Aws.amazon.com. 2021a. "¿Qué Es AWS IoT? - AWS IoT Core."
- Aws.amazon.com. 2021b. "AWS | Servicio de Base de Datos Gestionada NoSQL (DynamoDB)." Retrieved July 11, 2021 (https://aws.amazon.com/es/dynamodb/?trk=ps_a134p000006gXuVAAU&trkCampaigned=acq_paid_search_brand&sc_channel=PS&sc_campaign=acquisition_LATAMO&sc_publisher=Google&sc_category=Database&sc_country=LATAM O&sc_geo=LATAM&sc_outcome=acq&sc_detail=aws dynamodb&sc_content=DynamoDB_e&sc_matchtype=e&sc_segment=490481979011&sc_medium=ACQ-P%7CPS-GO%7CBrand%7CDesktop%7CSU%7CDatabase%7CDynamoDB%7CLATAM O%7CEN%7CText&s_kwid=ALi4422!3!490481979011!e!!g!!aws dynamodb&ef_id=Cj0KQCjwraqHBhDsARIsAKuGZeHL6vTi7hSvAtlhStSPTT UVMdSU_x_WH0KvVTEgvOGwvKDflE9sDcaAqJBEALw_wcB:G:s&s_kwid=ALi4422!3!490481979011!e!!g!!aws dynamodb).
- Bioalimentar.com. 2021. "La Temperatura En Pollitos - BiOALiMENTAR Ecuador." Retrieved July 2, 2021 (<https://www.bioalimentar.com/consejos-bio/la-temperatura-en-pollitos/>).
- Cedia. 2020. "Sector Avícola En El Ecuador." Retrieved February 6, 2021 (<https://blogs.cedia.org.ec/obest/wp-content/uploads/sites/7/2020/09/Sector-avicola-Ecuador.pdf>).
- Conocimientolibre.mx. 2021. "Tipos de Actuadores En IoT - Conocimiento Libre." Retrieved February 6, 2021 (<https://conocimientolibre.mx/tipos-de-actuadores-en-iot/>).
- ecured.cu. 2021. "Avicultura - EcuRed." Retrieved February 3, 2021 (<https://www.ecured.cu/Avicultura>).
- EcuRed. 2021. "Pollos de Engorde - EcuRed." Retrieved February 6, 2021 (https://www.ecured.cu/Pollos_de_engorde).

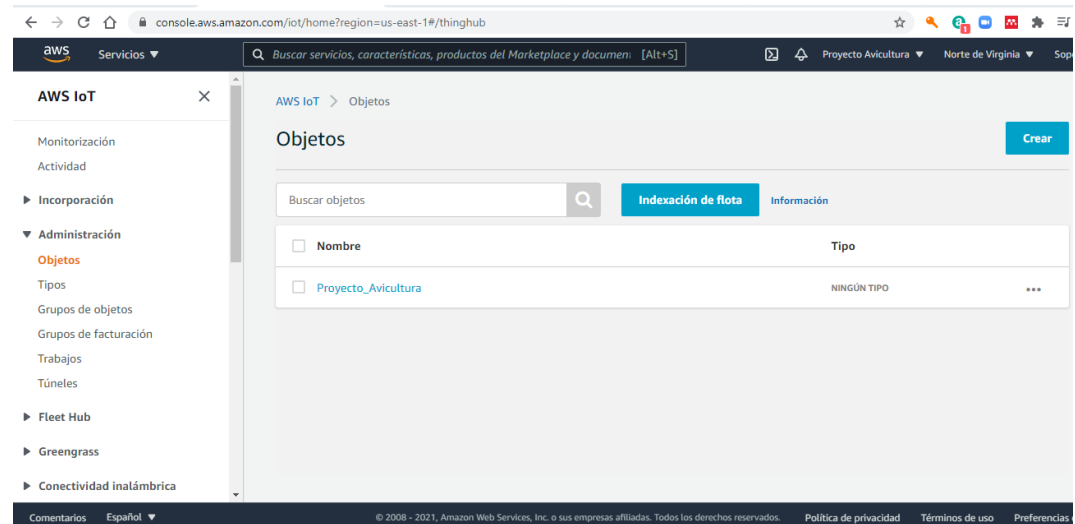
- Electronilab.co. 2021. "TSL2591 – Sensor de Intensidad de Luz Digital – Luxómetro - Electronilab." Retrieved February 6, 2021 (<https://electronilab.co/tienda/tsl2591-sensor-de-intensidad-de-luz-digital-luxometro/>).
- Elsitioavicola.com. 2012. "Control de Factores Ambientales En La Crianza de Pollitos: 1 - El Sitio Avicola." Retrieved February 5, 2021 (<https://www.elsitioavicola.com/articulos/2187/control-de-factores-ambientales-en-la-crianza-de-pollitos-1/>).
- Engormix.com. 2021. "Algoritmos y Avicultura: Una Combinación Poco Escuchada Pero Muy Efectiva - Engormix." Retrieved February 6, 2021 (<https://www.engormix.com/avicultura/articulos/algoritmos-avicultura-combinacion-poco-t45419.htm>).
- espressif.com. 2020. "ESP32 Wi-Fi & Bluetooth MCU | Espressif Systems." Retrieved October 22, 2020 (<https://www.espressif.com/en/products/socs/esp32>).
- Grafana.com. 2021. "Grafana: The Open Observability Platform | Grafana Labs."
- Hablemosdeaves.com. 2021. "¿Quieres Saber Todo Sobre Pollo de Engorde? Descúbrelo Aquí." Retrieved February 6, 2021 (<https://hablemosdeaves.com/pollo-de-engorde/>).
- Hiberus.com. 2020. "Amazon Web Services (AWS): ¿qué Es y Qué Ofrece? - Blog de Hiberus Tecnología." Retrieved October 23, 2020 (<https://www.hiberus.com/crecemos-contigo/amazon-web-services-aws-que-es-y-que-ofrece/>).
- iotbytes.wordpress.com. 2021. "Actuadores Básicos de IoT: Bytes de IoT." Retrieved February 6, 2021 (<https://iotbytes.wordpress.com/basic-iot-actuators/>).
- Luisllamas.es. 2021. "Medir La Humedad Del Suelo Con Arduino y Sensor FC-28." Retrieved February 6, 2021 (<https://www.luisllamas.es/arduino-humedad-suelo-fc-28/>).
- Ouali, Anis, Kin Fai Poon, Beum Seuk Lee, and Kaltham Al Romaihi. 2016. "Towards Achieving Practical GPON FTTH Designs." in *2015 IEEE 20th International Workshop on Computer Aided Modelling and Design of Communication Links and Networks, CAMAD 2015*.
- Planv.com.e. 2021. "El Pollo Nuestro de Cada Día: Los Impactos de La Industria de La Carne En El Ecuador | Plan V." Retrieved February 3, 2021 (<https://www.planv.com.ec/investigacion/investigacion/el-pollo-nuestro-cada-dia-impactos-la-industria-la-carne-el-ecuador>).
- Programarfacil.com. 2021. "Cómo Utilizar El DHT11 Para Medir La Temperatura y Humedad Con Arduino." Retrieved February 6, 2021 (<https://programarfacil.com/blog/arduino-blog/sensor-dht11-temperatura-humedad-arduino/>).
- Red Hat. 2020. "¿Qué Es El Internet de Las Cosas?" Retrieved October 22, 2020 (<https://www.redhat.com/es/topics/internet-of-things/what-is-iot>).
- Techlandia.com. 2021. "¿Cómo Puedo Crear Mi Propia Dirección de Correo Electrónico?" Retrieved February 6, 2021 (https://techlandia.com/crear-propia-direccion-correo-electronico-manera_86747/).
- Xiaomitoday.com. 2021. "6 Reasons Why You Should Choose Cloud Web Hosting! - XiaomiToday." Retrieved February 6, 2021 (<https://www.xiaomitoday.com/2018/05/26/cloud-web-hosting-benefits/>).

Guía de práctica de laboratorio

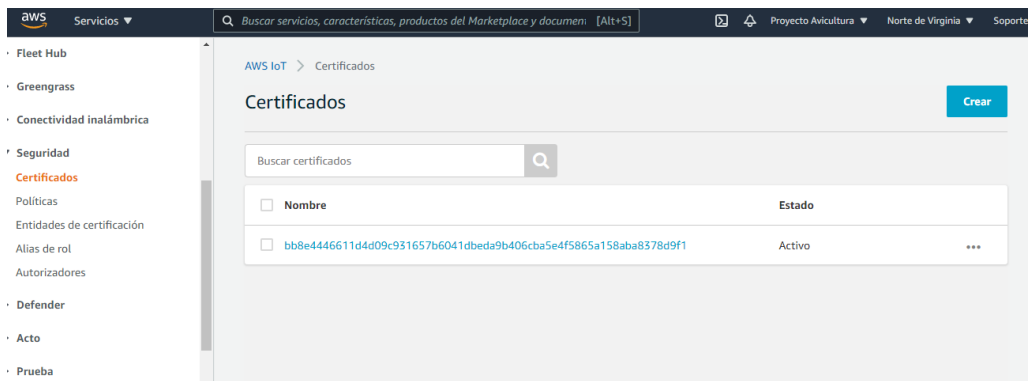
		GUÍA DE PRÁCTICA DE LABORATORIO
CARRERA: INGENIERÍA ELECTRÓNICA		ASIGNATURA:
NRO. PRÁCTICA:	1	TÍTULO PRÁCTICA: USO DEL PROTOTIPO IOT DE MONITORIZACION DE PARÁMETROS AMBIENTALES APLICADO A LA AVICULTURA EN GRANJAS DE POLLOS.
<p>OBJETIVO GENERAL:</p> <p>Configurar y validar el prototipo IoT aplicado a la avicultura</p> <p>OBJETIVOS ESPECÍFICOS:</p> <ul style="list-style-type: none"> • Tomar datos de los sensores del prototipo IoT de avicultura. • Analizar los datos del prototipo IoT aplicado a la avicultura. • Comprender las alarmas obtenidas del prototipo IoT aplicado a la avicultura. 		
INSTRUCCIONES		1. Armar grupos para el desarrollo de la práctica
		2. Los estudiantes previamente a la práctica deben leer en detalle esta guía de práctica.
		3. Los estudiantes deben usar los equipos y materiales para la práctica de una manera profesional siguiendo las normas del laboratorio y cuidando de los equipos, herramientas, así como los materiales indicados por el docente.
		4. Al finalizar la práctica el grupo de estudiantes debe dejar su sitio en orden y los equipos deben ser guardados en el laboratorio.
<p>ACTIVIDADES POR DESARROLLAR:</p> <p>En este manual de práctica se detalla el uso y configuración del prototipo IoT para realizar pruebas de laboratorio con los estudiantes de la carrera de ingeniería en telecomunicaciones.</p> <p>Monitoreo y configuraciones en AWS IOT</p> <p>AWS almacena los datos en la nube por medio de Kinesis Data Analytics que es la encargada de monitorear en tiempo real de los dispositivos IoT que son</p>		

implementados. También proporciona servicios automáticos para la recopilación, procesamiento, análisis y visualización de las actividades y los datos que proporciona los dispositivos IoT.

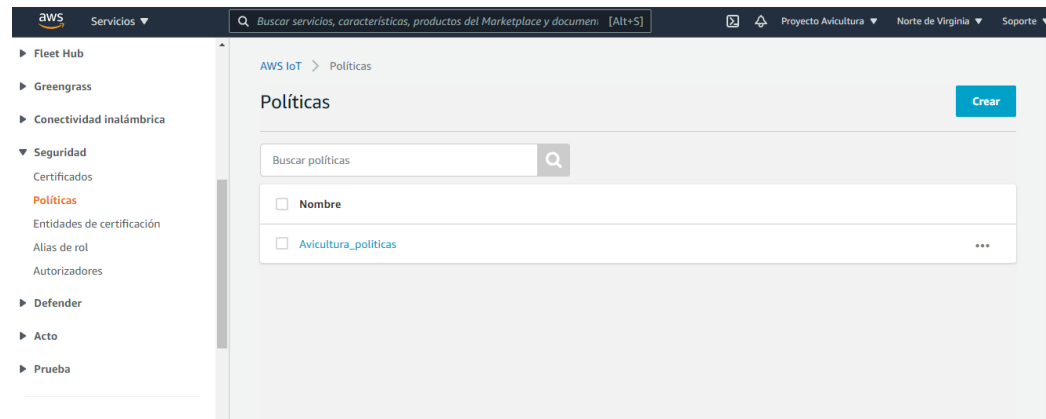
- 1) Se crea un objeto con el nombre del proyecto en este caso es "Proyecto Avicultura", en el apartado de Administración - Objetos que ayudara a monitorear y almacenar la información del prototipo IoT Avicultura.



- 2) A continuación se visualiza el apartado de Seguridad - Certificados que tiene el AWS IoT el cual proporciona seguridad al prototipo IoT avicultura.

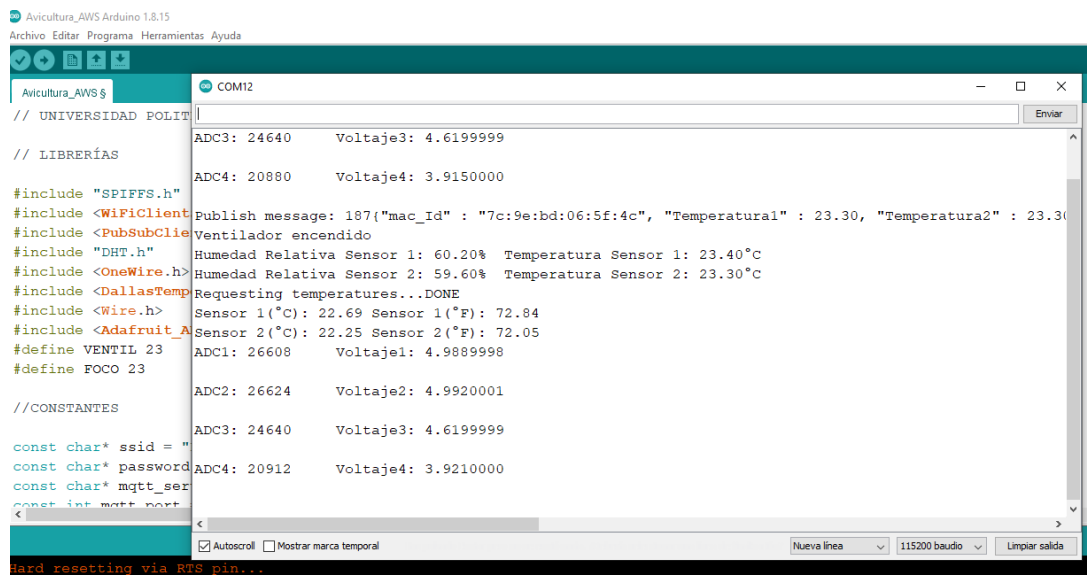


- 3) En este paso se visualiza el apartado de Seguridad – Política que tiene el AWS IoT el cual proporciona seguridad al prototipo IoT avicultura.



Monitoreo de sensores en IDE de Arduino

- 4) Para la monitorización de los sensores se utiliza el IDE de Arduino, donde se realiza el código para el funcionamiento de los sensores y poder así obtener resultados con cada uno de ellos como se muestra en la figura, donde se visualiza el voltaje, la temperatura, si el ventilador está activado, la humedad relativa de los sensores e intensidad lumínica.
- 5) Se debe agregar líneas de comando para vincular el enlace con AWS IOT y respaldar los datos en la nube. En la configuración de void setup () se añade el archivo SPIFFS para la lectura de archivos de AWS.



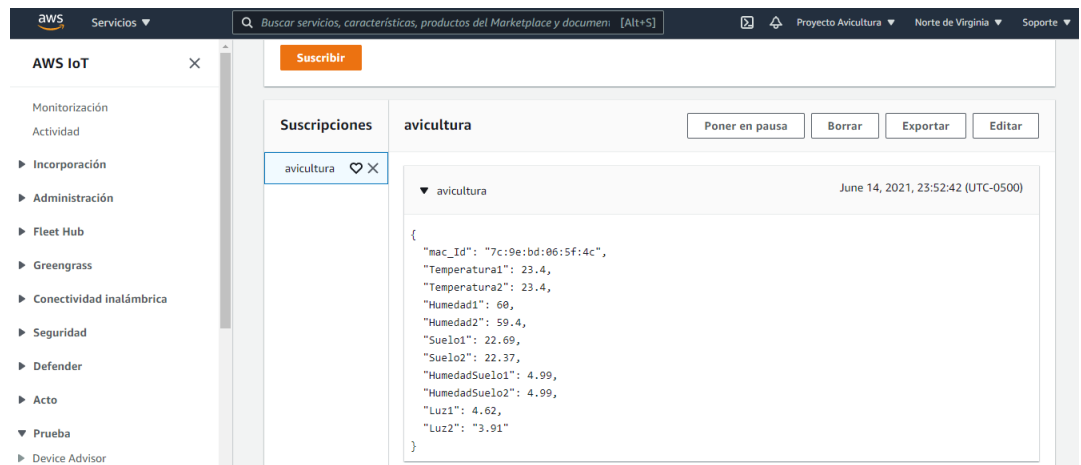
- 6) Una vez leída las cadenas que se le envía, se convertirán en punteros debido a que ESPClient no acepta cadenas, los cuales imprimen los punteros

convertidos para compararlos con el original y verificar los resultados.

- 7) Una vez estos punteros son convertidos a ESPCliente, se inicia el servidor MQTT END Point y el puerto. Siendo posible leer la dirección Mac del módulo ESP32.
- 8) Cuando se finaliza el conteo de cadena se leen las lecturas de temperatura y humedad de los sensores implementados y muestra el dispositivo vinculado. Se comprueba la conexión al cliente, en el caso de que no esté conectado, se llama a la función de reconexión para establecer la conexión.
- 9) Por último, se procede a convertir las variables que ingresan de los sensores a formato de cadena para volver a convertirlos al formato JSON, para que el ESP32 muestre por medio de parpadeos que la información llegó con éxito al AWS. Una vez ingresada toda la información en el IDE de Arduino se cargará a la placa de ESP32 para el funcionamiento de los sensores en el prototipo IoT Avicultura.

Monitor de sensores desde AWS

- 10) Para realizar el monitoreo de sensores en AWS, se debe crear un objeto en la administración en AWS IOT, una vez creado se generará un certificado y claves. Una vez que se genera, se procede a realizar las políticas que se definirán en AWS para los dispositivos IoT como se muestra en la figura.



- En el monitor serial se ve el ESP32 conectado al wifi programado y asignado con la dirección IP, luego lee los certificados de AWS de los SPIFFI y los

muestra, también muestra los certificados en punteros formato que está pasando al cliente. Lee el ID de la Mac de ESP32 y lo muestra, pasa a iniciar la conexión MQTT con la nube de AWS conectándose.

- A continuación, pasa a publicar datos JSON en el tema de MQTT, se observa la temperatura y la humedad recibida en el dispositivo conectado. Todos estos datos son visualizados en AWS, en el apartado de MQTT Client.
- Observando que todos los datos recibidos llegan a la nube de AWS mediante los sensores que se encuentran vinculados con ESP32. Recibiendo datos sin problemas y ningún tipo de error.

RESULTADO(S) OBTENIDO(S):

El estudiante debe colocar los resultados obtenidos en la práctica con imágenes y descripciones de cada imagen

CONCLUSIONES:

El estudiante debe colocar las conclusiones de las prácticas de acuerdo con los objetivos planteados.

RÚBRICA DE REVISIÓN DE PRÁCTICA:

Fecha de realización de la práctica:	
Integrantes del Grupo:	1.- 2.-

	Excelente	Muy Bueno	Bueno	Regular	Mal	Muy Mal	Observaciones
Calidad, orden, evidencia fotográfica y correcta estructuración del desarrollo de la práctica [30% del puntaje]							
Sustentación correcta de las prácticas [30% del puntaje]							
Conclusiones de las prácticas [40% del puntaje]							
PUNTAJE:							/10

Código de ESP32 en IDE Arduino

```
// DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO IOT PARA EL MONITOREO
DE PARÁMETROS AMBIENTALES
// APLICADOS A LA AVICULTURA PARA LA CRIANZA DE POLLOS DE GRANJA
UTILIZANDO HARDWARE DE BAJO COSTO Y AWS
// AUTORES: BRYAN GARCÍA - FÉLIX MORA
// UNIVERSIDAD POLITÉCNICA SALESIANA - INGENIERÍA ELECTRÓNICA
```

```
// LIBRERÍAS
```

```
#include "SPIFFS.h"
#include <WiFiClientSecure.h>
#include <PubSubClient.h>
#include "DHT.h"
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <Adafruit_ADS1015.h>
```

```
// RELAYS
```

```
#define VENTIL 23
#define FOCO 19
```

```
// GSM - SMS
```

```
const char simPIN[] = "";
#define SMS_TARGET "+593[REDACTED]"
#define TINY_GSM_MODEM_SIM800 // Modem is SIM800
#define TINY_GSM_RX_BUFFER 1024 // Set RX buffer to 1Kb
```

```
#include <TinyGsmClient.h>
```

```
#define MODEM_PWKEY 18
#define MODEM_TX 17
#define MODEM_RX 16
```

```
// Set serial for debug console (to Serial Monitor, default speed 115200)
```

```
#define SerialMon Serial
```

```
// Set serial for AT commands (to SIM800 module)
```

```
#define SerialAT Serial2
```

```
// Define the serial console for debug prints, if needed
```

```
//#define DUMP_AT_COMMANDS
```

```
#ifndef DUMP_AT_COMMANDS
```

```
#include <StreamDebugger.h>
```

```
StreamDebugger debugger(SerialAT, SerialMon);
```

```

    TinyGsm modem(debugger);
#else
    TinyGsm modem(SerialAT);
#endif

#define IP5306_ADDR      0x75
#define IP5306_REG_SYS_CTL0 0x00

//CONSTANTES

const char* ssid = "NETLIFE-GARCIA-VACA"; //Provide your SSID
const char* password = "XXXXXXXXXX";
const char* mqtt_server = "a1oidd02gc9iwf-ats.iot.us-east-1.amazonaws.com"; //
Relace with your MQTT END point
const int mqtt_port = 8883;

WiFiClientSecure espClient;
PubSubClient client(espClient);

String Read_rootca;
String Read_cert;
String Read_privatekey;
//=====
//=====
#define BUFFER_LEN 256
long lastMsg = 0;
char msg[BUFFER_LEN];
int value = 0;
byte mac[6];
char mac_Id[18];
int count = 1;
//=====
//=====

//UMBRALES - THRESHOLDS

float temperaturaThreshold = 24; // Límite de Temperatura - Lo normal debe estar
entre 24 y 28 grados no sobrepasar los 30
float humedadThreshold = 70; // Límite de Humedad relativa - Lo normal debe estar
entre 70 y 80%
float TemperaturaSueloThreshold = 24; // la temperatura del suelo depende de la
temperatura ambiente. Normal debe estar entre 24 y 28 grados.
float humedadSueloThreshold = 2; // Menor a 2 V la tierra esta humeda
float LDRThreshold = 4; // Mayor a 4 V es poca luz

//TEMP - HUMEDAD AMBIENTE

```

```

#define DHTPIN_1 13 // Digital pin connected to the DHT sensor
#define DHTPIN_2 14 // Digital pin connected to the DHT sensor
#define DHTTYPE DHT22 // DHT 22
DHT dht1(DHTPIN_1, DHTTYPE);
DHT dht2(DHTPIN_2, DHTTYPE);

// TEMP NIVEL DE SUELO
#define ONE_WIRE_BUS 15
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
DeviceAddress sensor1 = { 0x28, 0xCC, 0xC6, 0x75, 0xD0, 0x1, 0x3C, 0xAB };
DeviceAddress sensor2 = { 0x28, 0x82, 0xBE, 0x75, 0xD0, 0x1, 0x3C, 0x9E };

//SENSORES HUMEDAD DE SUELO - LUMINOSIDAD

Adafruit_ADS1115 ads(0x49);
float Voltage1 = 0.0;
float Voltage2 = 0.0;
float Voltage3 = 0.0;
float Voltage4 = 0.0;

// CONECTANDO A RED WIFI

void setup_wifi() {
  delay(10);
  Serial.println();
  Serial.print("Conectando a");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  randomSeed(micros());

  Serial.println("");
  Serial.println("WiFi conectado");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

// MQTT

void callback(char* topic, byte* payload, unsigned int length) {

```

```

    Serial.print("Message arrived [");
    Serial.print(topic);
    Serial.print("] ");
    for (int i = 0; i < length; i++) {
        Serial.print((char)payload[i]);
    }
    Serial.println();
}
void reconnect() {
    // Loop until we are reconnected
    while (!client.connected()) {
        Serial.print("Attempting MQTT connection...");
        // Create a random client ID
        String clientId = "ESP32-";
        clientId += String(random(0xffff), HEX);
        // Attempt to connect
        if (client.connect(clientId.c_str())) {
            Serial.println("connected");
            // Once connected, publish an announcement...
            client.publish("ei_out", "hello world");
            // ... and resubscribe
            client.subscribe("ei_in");
        } else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            // Wait 5 seconds before retrying
            delay(5000);
        }
    }
}

void setup() {

    // SENSORES

    Serial.begin(115200);

    Serial.println("DHTPIN_1 test!");
    Serial.println("DHTPIN_2 test!");

    dht1.begin();
    dht2.begin();

    sensors.begin();
    ads.begin();
    setup_wifi();

    // RELAYS

```

```

pinMode(VENTIL, OUTPUT);
pinMode(FOCO, OUTPUT);

// AWS

  Serial.setDebugOutput(true);
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(2, OUTPUT);
  setup_wifi();
  delay(1000);
  //=====
  if (!SPIFFS.begin(true)) {
    Serial.println("An Error has occurred while mounting SPIFFS");
    return;
  }
  //=====
  //Root CA File Reading.
  File file2 = SPIFFS.open("/AmazonRootCA1.pem", "r");
  if (!file2) {
    Serial.println("Failed to open file for reading");
    return;
  }
  Serial.println("Root CA File Content:");
  while (file2.available()) {
    Read_rootca = file2.readString();
    Serial.println(Read_rootca);
  }
  //=====
  // Cert file reading
  File file4 = SPIFFS.open("/bb8e444661-certificate.pem.crt", "r");
  if (!file4) {
    Serial.println("Failed to open file for reading");
    return;
  }
  Serial.println("Cert File Content:");
  while (file4.available()) {
    Read_cert = file4.readString();
    Serial.println(Read_cert);
  }
  //=====
  //Privatekey file reading
  File file6 = SPIFFS.open("/bb8e444661-private.pem.key", "r");
  if (!file6) {
    Serial.println("Failed to open file for reading");
    return;
  }
  Serial.println("privateKey File Content:");

```

```

while (file6.available()) {
  Read_privatekey = file6.readString();
  Serial.println(Read_privatekey);
}
//=====================================================

char* pRead_rootca;
pRead_rootca = (char *)malloc(sizeof(char) * (Read_rootca.length() + 1));
strcpy(pRead_rootca, Read_rootca.c_str());

char* pRead_cert;
pRead_cert = (char *)malloc(sizeof(char) * (Read_cert.length() + 1));
strcpy(pRead_cert, Read_cert.c_str());

char* pRead_privatekey;
pRead_privatekey = (char *)malloc(sizeof(char) * (Read_privatekey.length() + 1));
strcpy(pRead_privatekey, Read_privatekey.c_str());

Serial.println("=====
=====");
Serial.println("Certificates that passing to espClient Method");
Serial.println();
Serial.println("Root CA:");
Serial.write(pRead_rootca);

Serial.println("=====
=====");
Serial.println();
Serial.println("Cert:");
Serial.write(pRead_cert);

Serial.println("=====
=====");
Serial.println();
Serial.println("privateKey:");
Serial.write(pRead_privatekey);

Serial.println("=====
=====");

espClient.setCACert(pRead_rootca);
espClient.setCertificate(pRead_cert);
espClient.setPrivateKey(pRead_privatekey);

client.setServer(mqtt_server, mqtt_port);
client.setCallback(callback);

```



```

//=====
//=====
WiFi.macAddress(mac);
snprintf(mac_Id, sizeof(mac_Id), "%02x:%02x:%02x:%02x:%02x:%02x",
         mac[0], mac[1], mac[2], mac[3], mac[4], mac[5]);
Serial.print(mac_Id);

//=====
//=====

// GSM - SMS
delay(1000);

// Set modem reset, enable, power pins
pinMode(MODEM_PWKEY, OUTPUT);

// Power-On SIM800
digitalWrite(MODEM_PWKEY, LOW);
delay(800);
digitalWrite(MODEM_PWKEY, HIGH);
delay(1000);
digitalWrite(MODEM_PWKEY, LOW);

// Set GSM module baud rate and UART pins
SerialAT.begin(115200, SERIAL_8N1, MODEM_RX, MODEM_TX);
delay(3000);

// Restart SIM800 module, it takes quite some time
// To skip it, call init() instead of restart()

// modem.restart();
modem.init(); //if you do not need the complete restart
delay(100);

// Unlock your SIM card with a PIN if needed
if (strlen(simPIN) && modem.getSimStatus() != 3 ) {
  modem.simUnlock(simPIN);
}

}

void loop() {

delay(2000);

// TEMP Y HUMEDAD AMBIENTE

float h1 = dht1.readHumidity();

```

```

float t1 = dht1.readTemperature();
float h2 = dht2.readHumidity();
float t2 = dht2.readTemperature();

// Check if any reads failed and exit early (to try again).
if (isnan(h1) || isnan(t1)) {
  Serial.println(F("Falla DHT sensor 1!"));
  return;
}

// Check if any reads failed and exit early (to try again).
if (isnan(h2) || isnan(t2)) {
  Serial.println(F("Falla DHT sensor 2!"));
  return;
}

// Temp ambiente - Humedad Relativa
Serial.print(F("Humedad Relativa Sensor 1: "));
Serial.print(h1);
Serial.print(F("% Temperatura Sensor 1: "));
Serial.print(t1);
Serial.println(F("°C "));

Serial.print(F("Humedad Relativa Sensor 2: "));
Serial.print(h2);
Serial.print(F("% Temperatura Sensor 2: "));
Serial.print(t2);
Serial.println(F("°C "));

delay(2000);

// Temp Suelo
Serial.print("Requesting temperatures...");
sensors.requestTemperatures(); // Send the command to get temperatures
Serial.println("DONE");

Serial.print("Sensor 1(°C): ");
Serial.print(sensors.getTempC(sensor1));
Serial.print(" Sensor 1(°F): ");
Serial.println(sensors.getTempF(sensor1));

Serial.print("Sensor 2(°C): ");
Serial.print(sensors.getTempC(sensor2));
Serial.print(" Sensor 2(°F): ");
Serial.println(sensors.getTempF(sensor2));

// HUMEDAD DE SUELO - LUMINOSIDAD

```

```

delay(2000);

int16_t adc1;
int16_t adc2;
int16_t adc3;
int16_t adc4;

adc1 = ads.readADC_SingleEnded(0);
adc2 = ads.readADC_SingleEnded(1);
adc3 = ads.readADC_SingleEnded(2);
adc4 = ads.readADC_SingleEnded(3);

Voltage1 = adc1*(0.1875/1000); // V - HUMEDAD SUELO
Voltage2 = adc2*(0.1875/1000); // V - HUMEDAD SUELO
Voltage3 = adc3*(0.1875/1000); // V - LUMINOSIDAD
Voltage4 = adc4*(0.1875/1000); // V - LUMINOSIDAD

Serial.print("ADC1: ");
Serial.print(adc1);
Serial.print("\tVoltaje1: ");
Serial.println(Voltage1, 7);
Serial.println();
delay(1000);

Serial.print("ADC2: ");
Serial.print(adc2);
Serial.print("\tVoltaje2: ");
Serial.println(Voltage2, 7);
Serial.println();
delay(1000);

Serial.print("ADC3: ");
Serial.print(adc3);
Serial.print("\tVoltaje3: ");
Serial.println(Voltage3, 7);
Serial.println();
delay(1000);

Serial.print("ADC4: ");
Serial.print(adc4);
Serial.print("\tVoltaje4: ");
Serial.println(Voltage4, 7);
Serial.println();
delay(1000);

```

```

if (!client.connected()) {
    reconnect();
}
client.loop();

long now = millis();
if (now - lastMsg > 2000) {
    lastMsg = now;

//=====
=====
String macIdStr = mac_Id;
String Temperatura1 = String(t1);
String Temperatura2 = String(t2);
String Humedad1 = String(h1);
String Humedad2 = String(h2);
String Suelo1 = String(sensors.getTempC(sensor1));
String Suelo2 = String(sensors.getTempC(sensor2));
String HumedadSuelo1 = String(Voltage1);
String HumedadSuelo2 = String(Voltage2);
String Luz1 = String(Voltage3);
String Luz2 = String(Voltage4);

    snprintf (msg, BUFFER_LEN, "{\"mac_Id\" : \"%s\", \"Temperatura1\" : %s,
    \"Temperatura2\" : %s, \"Humedad1\" : %s,\"Humedad2\" : %s,\"Suelo1\" :
    %s,\"Suelo2\" : %s,\"HumedadSuelo1\" : %s,\"HumedadSuelo2\" : %s,\"Luz1\" :
    %s,\"Luz2\" : \"%s\"}", macIdStr.c_str(), Temperatura1.c_str(), Temperatura2.c_str(),
    Humedad1.c_str(), Humedad2.c_str(), Suelo1.c_str(), Suelo2.c_str(),
    HumedadSuelo1.c_str(), HumedadSuelo2.c_str(), Luz1.c_str(), Luz2.c_str());
    Serial.print("Publish message: ");
    Serial.print(count);
    Serial.println(msg);
    client.publish("avicultura", msg);
    count = count + 1;

//=====
=====
}

delay(100);

// CONFIGURACIONES DE UMBRALES
// CONFIGURACIONES DE UMBRALES TEMPERATURA Y HUMEDAD

if (t1 > temperaturaThreshold){
    digitalWrite(VENTIL, HIGH);
    delay(1000);
    Serial.println("Ventilador encendido");
}

```

```

    }
else
{
    digitalWrite(VENTIL, LOW);
    delay(1000);
    Serial.println("Ventilador apagado");
}

delay(100);

// CONFIGURACIONES DE UMBRALES LUZ

if (Voltage3 > LDRThreshold){
    digitalWrite(FOCO, HIGH);
    delay(1000);
    Serial.println("Foco encendido");
}
else
{
    digitalWrite(FOCO, LOW);
    delay(1000);
    Serial.println("Foco apagado");
}

delay(100);

// ENVÍO DE SMS - THRESHOLDS
// CONFIGURACIONES DE UMBRALES TEMPERATURA AMBIENTE Y HUMEDAD
RELATIVA

// TEMPERATURA AMBIENTE
if (t1 > temperaturaThreshold ){
    String smsMessage = String ("Temperatura ambiente esta elevandose(Temp1): ") +
String (t1) + String (" C");
    if(modem.sendSMS(SMS_TARGET, smsMessage)){
        Serial.println("Mensaje enviado Temperatura ambiente (Temp1)");
    }
}
else if (t2 > temperaturaThreshold ){
    String smsMessage = String ("Temperatura ambiente esta elevandose (Temp2): ") +
String (t2) + String (" C");
    if(modem.sendSMS(SMS_TARGET, smsMessage)){
        Serial.println("Mensaje enviado Temperatura ambiente (Temp2)");
    }
}
else
{
    Serial.println("SMS no enviado Temperatura");
}

```

```

delay(100);

// HUMEDAD RELATIVA
if (h1 > humedadThreshold ){
  String smsMessage = String ("Humedad relativa esta elevandose(Hum1): ") + String
(h1) + String (" %");
  if(modem.sendSMS(SMS_TARGET, smsMessage)){
    Serial.println("Mensaje enviado Humedad realativa (Hum1)");
  }
}
else if (h2 > humedadThreshold ){
  String smsMessage = String ("Humedad relativa esta elevandose(Hum2): ") + String
(h2) + String (" %");
  if(modem.sendSMS(SMS_TARGET, smsMessage)){
    Serial.println("Mensaje enviado Humedad relativa (Hum2)");
  }
}
else
{
  Serial.println("SMS no enviado Humedad");
}
delay(100);

// TEMPERATURA SUELO

if ((sensors.getTempC(sensor1)) > TemperaturaSueloThreshold ){
  String smsMessage = String ("Temperatura suelo esta elevandose(sensor1): ") +
String ((sensors.getTempC(sensor1))) + String (" C");
  if(modem.sendSMS(SMS_TARGET, smsMessage)){
    Serial.println("Mensaje enviado Temperatura sensor1");
  }
}
else if ((sensors.getTempC(sensor2)) > TemperaturaSueloThreshold ){
  String smsMessage = String ("Temperatura suelo esta elevandose (sensor2): ") +
String ((sensors.getTempC(sensor1))) + String (" C");
  if(modem.sendSMS(SMS_TARGET, smsMessage)){
    Serial.println("Mensaje enviado Temperatura sensor2");
  }
}
else
{
  Serial.println("SMS no enviado Temperatura suelo");
}
delay(100);

// Humedad Suelo

if (Voltage1 > humedadSueloThreshold ){
  String smsMessage = String ("Nivel de agua bajo (sensor1)");

```

```

    if(modem.sendSMS(SMS_TARGET, smsMessage)){
        Serial.println("Mensaje enviado Nivel de agua 1");
    }
}
else if (Voltage2 > humedadSueloThreshold ){
    String smsMessage = String ("Nivel de agua bajo (sensor2)");
    if(modem.sendSMS(SMS_TARGET, smsMessage)){
        Serial.println("Mensaje enviado Nivel de agua 2");
    }
}
else
{
    Serial.println("SMS no enviado Nivel de agua");
}
delay(100);

// Luminosidad en maqueta

if (Voltage3 > LDRThreshold ){
    String smsMessage = String ("Hay poca luz en maqueta (sensor1)");
    if(modem.sendSMS(SMS_TARGET, smsMessage)){
        Serial.println("Mensaje enviado LDR1");
    }
}
else if (Voltage4 > LDRThreshold ){
    String smsMessage = String ("Hay poca luz en maqueta (sensor2)");
    if(modem.sendSMS(SMS_TARGET, smsMessage)){
        Serial.println("Mensaje enviado LDR2");
    }
}
else
{
    Serial.println("SMS no enviado LDR");
}
delay (600000);
}

```