



**UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD DE INGENIERÍA
CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
MONITOREO Y CONTROL AMBIENTAL BASADO EN IOT
PARA OPTIMIZAR LA PRODUCCIÓN DE HUEVOS**

**Trabajo de Titulación para optar al título de:
Ingeniero en Electrónica y Telecomunicaciones.**

Autor:

Obando Morillo Jonathan Fabian

Tutor:

Mgs. Alejandra del pilar Pozo Jara

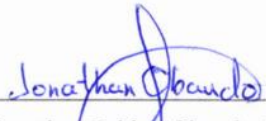
Riobamba, Ecuador. 2023

DERECHOS DE AUTORÍA

Yo, **Jonathan Fabian Obando Morillo**, con cédula de ciudadanía **0401620166**, autor del trabajo de investigación titulado **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO Y CONTROL AMBIENTAL BASADO EN IOT PARA OPTIMIZAR LA PRODUCCIÓN DE HUEVOS”**, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mi exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor (a) de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, 5 de junio de 2023.


Jonathan Fabian Obando Morillo
C.I.0401620166

DICTAMEN FAVORABLE DEL TUTOR Y MIEMBROS DE TRIBUNAL

Quienes suscribimos, catedráticos designados Tutor y Miembros del Tribunal de Grado para la evaluación del trabajo de investigación **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO Y CONTROL AMBIENTAL BASADO EN IOT PARA OPTIMIZAR LA PRODUCCIÓN DE HUEVOS”**, presentado por Jonathan Fabian Obando Morillo, con cédula de identidad número 0401620166, certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha asesorado durante el desarrollo, revisado y evaluado el trabajo de investigación escrito y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba 05 de junio de 2023

Dr. Ciro Diego Radicelli Garcia
PRESIDENTE DEL TRIBUNAL DE GRADO




PhD. Leonardo Fabian Rentería Bustamante
MIEMBRO DEL TRIBUNAL DE GRADO



PhD. Manuel Antonio Meneses Freire
MIEMBRO DEL TRIBUNAL DE GRADO



Mgs. Alejandra del Pilar Pozo Jara
TUTOR

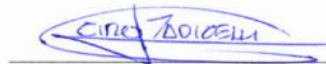


CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO Y CONTROL AMBIENTAL BASADO EN IOT PARA OPTIMIZAR LA PRODUCCIÓN DE HUEVOS.”** presentado por Jonathan Fabian Obando Morillo, con cédula de identidad número 0401620166 bajo la tutoría de Mgs. Alejandra del Pilar Pozo Jara; certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba 05 de junio de 2023.

Presidente del Tribunal de Grado
Dr. Ciro Diego Radicelli Garcia



Miembro del Tribunal de Grado
PhD. Leonardo Fabian Rentería Bustamante



Miembro del Tribunal de Grado
PhD. Manuel Antonio Meneses Freire





CERTIFICACIÓN

Que, **OBANDO MORILLO JONATHAN FABIAN** con CC: **0401620166**, estudiante de la Carrera **ELECTRÓNICA Y TELECOMUNICACIONES, NO VIGENTE**, Facultad de **INGENIERÍA** ha trabajado bajo mi tutoría el trabajo de investigación titulado **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO Y CONTROL AMBIENTAL BASADO EN IOT PARA OPTIMIZAR LA PRODUCCIÓN DE HUEVOS”**, cumple con el 3 %, de acuerdo al reporte del sistema Anti plagio **URKUND**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 30 de mayo de 2023



Firma de digitalización por:
**ALEJANDRA DEL PILAR
POZO JARA**

Mgs. Alejandra Pozo
TUTOR (A)

DEDICATORIA

El presente trabajo de titulación va dedicado primeramente a Dios, por darme vida para salir adelante, por permitirme llegar a este momento tan anhelado, en mi vida profesional y darme fuerza en momentos de adversidades y brindarme la sabiduría necesaria para el cumplimiento de cada una de mis metas.

A mis padres que con sus palabras siempre me animaron a seguir adelante, a mi hijo por ser mi inspiración, mi mayor motivo para cumplir mi sueño, y a toda mi familia por siempre regalarme una palabra de aliento y así motivarme a cumplir mis metas.

Jonathan Obando Morillo

AGRADECIMIENTO

En primer lugar, quiero expresar mi más sincero agradecimiento a la noble institución como es la Universidad Nacional de Chimborazo, por haberme brindado su estadía en sus aulas de clases, como un digno estudiante de tan reconocida institución.

A mi tutora la Mgs. Alejandra Pozo que me supo guiar en todo momento, por su valioso tiempo y su apoyo; agradezco a cada uno de los docentes que fueron partícipes de mi formación profesional y que cada día entregaron sus conocimientos y experiencias, que son de gran importancia para mi vida profesional.

A mis padres quienes han sido los pilares fundamentales de mi vida, agradezco su infinita paciencia, su sabio consejo y su dedicación inquebrantable. Han sido modelos de valentía, perseverancia y amor incondicional. Cada sacrificio que han hecho por mí no ha pasado desapercibido y siempre estaré agradecido por todo lo que han hecho y siguen haciendo por mí.

A mi esposa e hijo quienes fueron partícipes de este proceso, siempre han estado a mi lado, dándome ánimo y recordándome que puedo superar cualquier obstáculo, siendo su presencia un faro de luz en los momentos más difíciles

Jonathan Obando Morillo

ÍNDICE GENERAL

DERECHOS DE AUTORÍA.....	
DICTAMEN FAVORABLE DEL TUTOR Y MIEMBROS DE TRIBUNAL	
CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL	
CERTIFICADO ANTIPLAGIO	
DEDICATORIA.....	
AGRADECIMIENTO	
ÍNDICE GENERAL	
ÍNDICE DE TABLAS.....	
ÍNDICE DE FIGURAS	
RESUMEN	
ABSTRACT	
INTRODUCCIÓN.....	16
CAPÍTULO I.....	17
1 PLANTEAMIENTO DEL PROBLEMA	17
1.1 PROBLEMA Y JUSTIFICACIÓN.....	17
1.2 OBJETIVOS.....	18
1.2.1 GENERAL	18
1.2.2 ESPECÍFICOS	18
CAPÍTULO II.....	19
2 MARCO TEÓRICO.....	19
2.1 ANTECEDENTES.....	19
2.2 INCUBACIÓN ARTIFICIAL.....	19
2.2.1 MÉTODO DE ETAPA ÚNICA	20
2.2.2 MÉTODO DE ETAPA MÚLTIPLE	20
2.3 INCUBADORAS	20
2.3.1 INCUBADORAS MANUALES.....	20
2.3.2 INCUBADORAS AUTOMÁTICAS	21
2.4 MANEJO DE LA INCUBACIÓN	21
2.5 FACTORES CLAVE DE LA INCUBACIÓN	22
2.5.1 LA TEMPERATURA	22
2.5.2 LA HUMEDAD	23

2.5.3	MOVIMIENTO	23
2.5.4	VENTILACIÓN	24
2.6	COMPONENTES DE UNA INCUBADORA.....	25
2.6.1	ESTRUCTURA	25
2.6.2	CONTROL DE TEMPERATURA Y HUMEDAD	27
2.6.3	CONTROL DE MOVIMIENTO Y VENTILACIÓN.....	27
2.7	IOT	27
2.7.1	ARQUITECTURA DE IOT	28
2.7.2	PROTOCOLOS DE IOT	28
2.8	VISIÓN ARTIFICIAL	29
CAPÍTULO III		31
3	METODOLOGÍA	31
3.1	TIPO DE INVESTIGACIÓN	31
3.1.1	INVESTIGACIÓN NO EXPERIMENTAL.....	31
3.2	DISEÑO DE INVESTIGACIÓN	31
3.2.1	MÉTODOS DE INVESTIGACIÓN	31
3.3	TÉCNICAS DE RECOLECCIÓN DE DATOS	31
3.4	POBLACIÓN DE ESTUDIO Y TAMAÑO DE MUESTRA.....	31
3.4.1	POBLACIÓN	31
3.4.2	MUESTRA	31
3.5	HIPÓTESIS	32
3.5.1	OPERACIONALIZACIÓN DE VARIABLES.....	32
3.6	MÉTODO DE ANÁLISIS	32
3.7	PROCESAMIENTO DE DATOS.....	33
3.7.1	ANÁLISIS DE FACTIBILIDAD.....	34
3.8	REQUERIMIENTOS DEL PROTOTIPO	34
3.8.1	SELECCIÓN DE HARDWARE.....	34
3.8.2	DISEÑO DEL PROTOTIPO.....	37
3.9	IMPLEMENTACIÓN DEL PROTOTIPO	49
CAPÍTULO IV		54
4	RESULTADOS Y DISCUSIÓN	54
4.1	PRIMER PROCESO DE INCUBACIÓN	54
4.1.1	SELECCIÓN DE HUEVOS FÉRTILES	54

4.1.2	INICIO DEL PROCESO DE INCUBACIÓN	55
4.1.3	PRUEBAS DE OVOSCOPIA	55
4.1.4	FIN DE LA INCUBACIÓN	56
4.1.5	ANÁLISIS DE LOS PARÁMETROS DE HUMEDAD Y TEMPERATURA 57	
4.1.6	ANÁLISIS DE RESULTADOS.....	59
4.2	SEGUNDO PROCESO DE INCUBACIÓN	61
4.2.1	SELECCIÓN DE HUEVOS FÉRTILES	61
4.2.2	INICIO DEL PROCESO DE INCUBACIÓN	61
4.2.3	PRUEBAS DE OVOSCOPIA	62
4.2.4	FIN DE LA INCUBACIÓN	63
4.2.5	ANÁLISIS DE LOS PARÁMETROS DE HUMEDAD Y TEMPERATURA 64	
4.2.6	ANÁLISIS DE RESULTADOS.....	66
4.3	ANÁLISIS GENERAL	67
CAPÍTULO V. CONCLUSIONES y RECOMENDACIONES		70
5.1	CONCLUSIONES	70
5.2	RECOMENDACIONES	71
BIBLIOGRAFÍA		72
ANEXOS		75
Anexo 1. Datos de Temperatura y humedad por día de la incubadora HDD mini 56S ...		75
Anexo 2. Datos de Temperatura y humedad por día del prototipo.		80
Anexo 3. Resultados de los procesos de incubación.....		85
Anexo 4. Pruebas de Ovoscopia.....		86
Anexo 5. Programación en Arduino.....		87
Anexo 6. Programación en Python.....		91
Anexo 7. Programación de la aplicación Android		98

ÍNDICE DE TABLAS

Tabla 1. Temperatura Óptima de Incubación para Diferentes tipos de Aves.....	22
Tabla 2. Rangos de Humedad, para diferentes tipos de aves.	23
Tabla 3. Análisis de las variables dependiente e independientes	32
Tabla 4. Características de los Dispositivos Embebidos	35
Tabla 5. Características Sensor AHT10.	36
Tabla 6. Características Servomotor y Motor Bipolar.....	37
Tabla 7. Simulación de valores para Kp y Ti del Controlador PID	41
Tabla 8. Parámetros para el giro de las Regletas de Huevos.	51
Tabla 9. Parámetros para el movimiento del sistema mecánico encargado de desplazar la cámara a lo largo de la bandeja de huevos - Eje Y.....	52
Tabla 10. Parámetros para el movimiento del sistema mecánico encargado de desplazar la cámara a lo largo de la bandeja de huevos - Eje X.....	52
Tabla 11. Resultados de la Ovoscopia realizada en los días 9,14 y 18.	55
Tabla 12. Datos de temperatura y humedad obtenidos durante la incubación.	57
Tabla 13. Valores característicos para el análisis de la temperatura.	58
Tabla 14. Valores característicos para el análisis de la humedad.....	58
Tabla 15. Resultados del rendimiento en la producción de la incubadora.	60
Tabla 16. Resumen de los valores de Temperatura y Humedad obtenidos en la incubación.	64
Tabla 17. Resultados estadísticos de la humedad.....	64
Tabla 18. Resultados estadísticos humedad.	65
Tabla 19. Resultados del rendimiento en la producción del Prototipo.....	66
Tabla 20. Resultados del test de normalidad para el sistema existente.	67
Tabla 21. Resultados del test de normalidad para el sistema implementado.	68
Tabla 22. Resultados del test de Wilcoxon	69

ÍNDICE DE FIGURAS

Figura 1. Incubadora manual Covatutto Mod.16 de 12 huevos.	21
Figura 2. Incubadora automática HDD mini 56S para 55 huevos.	21
Figura 3. Principio de Ventilación Estática.	24
Figura 4. Principio de Ventilación Forzada.	25
Figura 5. Incubadora Masalles Mod.2600-I HLC con estructura de aluminio.	26
Figura 6. Incubadora de huevos con estructura de madera.	26
Figura 7. Controlador de temperatura y humedad.	27
Figura 8. Arquitectura de una Red IoT y protocolos Disponibles.	28
Figura 9. Flujograma del desarrollo del proyecto.	33
Figura 10. Dispositivos embebidos, Arduino Mega 2560 y Raspberry Pi 4	35
Figura 11. A) Sensor AHT10. B) Cámara Logitech B525	36
Figura 12. A) Niquelina Eléctrica, B) Motor NEMA 17, C) Servo motor DS3235 D) Ventilador PWM	36
Figura 13. A) Cámara de incubación. B) Cubeta para los Huevos. C) Estructura de Desplazamiento	38
Figura 14. Diagrama general del Controlador PID a implementar.	38
Figura 15. Curva de comportamiento del sistema a un Cambio escalón.	39
Figura 16. Curva de comportamiento del sistema en variables de Desviación	39
Figura 17. Selección de τ con respecto a los datos obtenidos del sistema	40
Figura 18. Respuesta de $y(t)$ al sistema.	41
Figura 19. Modelado en 3D del sistema mecánico a implementar.	43
Figura 20. Diagrama de conexiones para el mecanismo de movimiento.	44
Figura 21. Algoritmo para el Proceso de Ovoscopia.	44
Figura 22. Algoritmo aplicado para la Detección de anomalías en los huevos.	45
Figura 23. Diagrama de constitución de los dispositivos en el sistema IoT a implementar.	46
Figura 24. Pantalla en Kivy de Usuario y Gráficos	47
Figura 25. Pantalla en Kivy de Cámara y Registro.	47
Figura 26. Pantalla en Kivy de Notificaciones y Configuración	48
Figura 27. Pantalla de Visualización principal.	48
Figura 28. Interfaz Gráfica Remota (Dispositivo Android).	49
Figura 29. Armado de la Cámara de incubación, Cubeta para los Huevos y Estructura de Desplazamiento.	49
Figura 30. Prueba de funcionamiento del sistema de otoscopia.	50
Figura 31. Sistema de movimientos de Huevos.	50
Figura 32. Sistema de Desplazamiento de la cámara.	51
Figura 33. Conexión de los sistemas implementados en el proyecto (Simulador Proteus).	53
Figura 34. Colocación de los huevos fértiles en la incubadora.	54
Figura 35. Encendido de la incubadora e inicio del Proceso de Incubación.	55
Figura 36. Resultados. Mortalidad en etapa inicial y media del proceso de incubación.	56
Figura 37. Resultados, Mortalidad en etapa final del proceso de incubación.	56
Figura 38. Gráfica de Temperatura del proceso de incubación.	59

Figura 39. Gráfica de Humedad del proceso de Incubación.	59
Figura 40. Curva de rendimiento del sistema.....	60
Figura 41. Grafica del rendimiento del sistema.....	60
Figura 42. Colocación de los huevos fértiles en el prototipo Implementado.	61
Figura 43. Encendido y puesta en funcionamiento del prototipo.....	62
Figura 44. Primera prueba de ovoscopia en el día 7. A) Huevo Fértil. B) Huevo Infértil. .	62
Figura 45. Segunda prueba de ovoscopia en el día 14. A) Huevo Fértil. B) Huevo Infértil.	62
Figura 46. Ultima prueba de ovoscopia en el día 18. A) Huevo Fértil. B) Huevo Infértil..	63
Figura 47. Eclosión de huevos.....	63
Figura 48. Polluelos obtenidos al final de la incubación.....	63
Figura 49. Grafica del comportamiento de la Temperatura.....	65
Figura 50. Grafica del comportamiento de la Humedad.	66
Figura 51. Grafica de la Curva de Rendimiento del Prototipo.	66
Figura 52. Gráfico de Rendimiento del Prototipo.	67
Figura 53. Ovoscopia realizada a los huevos de la incubadora HDD mini 56S.....	86
Figura 54. Ovoscopia realizada a los huevos del prototipo.....	86

RESUMEN

El sistema implementado para mejorar la producción de huevos de aves de corral se basa en el monitoreo y control ambiental a través de IoT. Este sistema utiliza materiales disponibles en el mercado y consta de un microcontrolador que captura y procesa las variables ambientales, las cuales desempeñan un papel crucial en la producción de huevos. Además, se incorpora una microcomputadora que proporciona servicios adicionales no presentes en una incubadora tradicional, permitiendo la conexión a internet y convirtiéndola en un dispositivo IoT que puede ser monitoreado y controlado de forma remota.

El proyecto se desarrolla utilizando una metodología no experimental, lo que permite observar los fenómenos que ocurren durante el proceso de incubación sin intervenir directamente en el objeto de estudio. Para recopilar los datos se emplea la técnica de observación, que ayuda a identificar el comportamiento del objeto de estudio de manera no intrusiva, en dos muestras realizadas con una población de 55 huevos de gallina, se demuestra que el prototipo desarrollado obtuvo una mayor productividad, logrando una eficiencia de incubación del 30% en comparación con el 0% de la incubadora HDD mini 56S.

En resumen, el sistema implementado, basado en IoT, muestra mejoras significativas en la producción de huevos de aves de corral. Los resultados obtenidos en las pruebas realizadas demuestran la eficiencia del prototipo desarrollado en comparación con una incubadora convencional, destacando su capacidad de monitoreo y control remoto a través de una conexión a Internet.

Palabras claves: IoT, Sistema Automático, Producción, Monitoreo, Control

Abstract

The system implemented to improve poultry egg production is based on environmental monitoring and control through IoT. This system uses commercially available materials and consists of a microcontroller that captures and processes environmental variables, which play a crucial role in egg production. In addition, a microcomputer is incorporated that provides additional services that traditional incubators do not have. This allows Internet connection and turning it into an IoT device that can be monitored and controlled remotely.

The project is developed using a non-experimental methodology, which allows observing the phenomena that occur during the incubation process without directly intervening in the object of study. To collect the data, the observation technique is used, which helps to identify the behavior of the object of study in a non-intrusive way, in two samples made with a population of 55 chicken eggs, it is shown that the developed prototype obtained greater productivity, achieving 30% incubation efficiency compared to 0% for HDD mini 56S incubator.

In summary, the implemented system, based on IoT, shows significant improvements in poultry egg production. The results obtained in the tests carried out demonstrate the efficiency of the developed prototype in comparison with a conventional incubator, highlighting its monitoring capacity and remote control through an Internet connection.

Keywords: IoT, Automatic System, Production, Monitoring, Control



Elmado electrónicamente por:
GABRIELA MARIA DE
LA CRUZ FERNANDEZ

Reviewed by:

Gabriela de la Cruz F. Msc

ENGLISH PROFESSOR

C.C. 0603467929

INTRODUCCIÓN

Con el desarrollo de la tecnología y la automatización en la industria se puede llegar a obtener grandes beneficios como la reducción de tiempos de producción, mejoras en la calidad del producto y prevenir riesgos para las personas en ambientes peligrosos, siendo esta ampliamente implementada en campos como: alimentación, construcción, educación, agrícola y automotriz por mencionar algunos.

Por otra parte, el uso del Internet de las cosas (IoT por sus siglas en inglés) en el sector agropecuario como una herramienta tecnológica, permite evaluar variables ambientales mediante el uso de sensores en sistemas como: invernaderos [1], granjas [2], e incluso facilitando el monitoreo del estado de salud en animales [3].

Si bien es cierto que la incubación artificial de huevos es una técnica que se viene practicando desde la antigüedad, con el avance de la tecnología se ha conseguido proporcionar un ambiente adecuado para la incubación de huevos, mediante la implementación de sistemas de control y monitoreo, que permiten mantener la temperatura [4], humedad, ventilación e incluso realizar una selección de huevos que no se encuentren fertilizados [5].

En el presente trabajo se desarrollará una incubadora en la cual se implementará un sistema de control el que se encargará de controlar la temperatura, humedad, ventilación y el giro de los huevos para su óptimo desarrollo, y un sistema de monitoreo remoto para obtener imágenes en tiempo real del desarrollo de los mismos, además de hacer posible la selección temprana de aquellos que no se encuentren aptos para completar el proceso de incubación, siendo esta información enviada a un servidor para ser procesada y a su vez enviada al operador haciendo uso de protocolos y procedimientos del internet de las cosas (IoT), también se desarrollará una aplicación que permitirá tanto el control local como remoto de la incubadora.

CAPÍTULO I.

1 PLANTEAMIENTO DEL PROBLEMA

1.1 PROBLEMA Y JUSTIFICACIÓN

En Ecuador el sector avícola es parte de la cadena productiva de maíz, soja, y balanceados, siendo esta una de las de mayor importancia dentro del sector agropecuario Ecuatoriano [6], y según el censo realizado por el instituto de estadística y censos (INEC) en 2020, los planteles avícolas tuvieron una producción de 81.73 millones de huevos de gallinas, a comparación de los 4.70 millones de huevos producidos por gallinas criadas en campo, siendo Tungurahua el sector donde existe la mayor producción con 32.28 millones de huevos [7].

Por lo cual, en la actualidad los pequeños productores se encuentran ante un escenario complejo, enfrentándose a un ambiente más competitivo, para lo cual estos deben desarrollar estrategias y mecanismos que les permita dinamizar la producción, optando por modernizar los procesos de incubación y cría de pollos y gallinas, para lo cual los procesos de control y la constante monitorización son la principal prioridad para lograr un desarrollo óptimo de estos.

Además, en el proceso de incubación existen varios factores que afectan a la misma, como pueden ser: la mala ubicación de los huevos en las bandejas, siendo esto fatal para el embrión, pues este se desarrolla con la cabeza hacia la parte superior del huevo y su primera respiración la hace a través de la cámara de aire la cual puede o no estar allí dependiendo de la correcta colocación; Otro de los problemas que se puede encontrar es que los huevos se encuentren rotos para lo cual se debe de realizar una inspección minuciosa antes de ser colocados en las incubadoras; y otro inconveniente que es poco común es el estallido de huevos, lo cual afecta el ambiente dentro de la incubadora llegando a dañar toda la producción y afectando al avicultor.

En el presente trabajo de investigación se pretende diseñar una incubadora de huevos capaz de solventar los problemas anteriormente mencionados, la cual estará compuesta de un sistema de control a través de sensores eléctricos de temperatura, humedad y ventilación generando un manejo óptimo del ambiente en su interior, para ello la temperatura debe encontrarse entre 37.5 °C a 37.7 °C durante la primera etapa de incubación (18 días) y de 36.5 °C a 37 °C, en los últimos 3 días, por otro lado, la humedad debe mantenerse entre un 55 a 66 % en la primera etapa y de 70 a 75 % en la segunda [8] además el sistema permitirá un monitoreo inteligente basado en una cámara digital con la cual se detectará e informará de anomalías dentro de la incubadora sin tener la intervención directa en esta.

1.2 OBJETIVOS

1.2.1 GENERAL

- Diseñar e implementar un sistema de monitoreo y control ambiental basado en IoT para la optimización en la producción de huevos.

1.2.2 ESPECÍFICOS

- Diseñar un sistema de control y monitoreo automatizado para el control de variables ambientales en una incubadora y monitoreo del estado de los huevos durante el proceso de incubación.
- Definir los algoritmos de visión artificial para el monitoreo inteligente y seleccionar un protocolo IoT para la conexión y almacenamiento de datos en la nube.
- Implementar un prototipo de incubadora que cumpla con los parámetros del sistema propuesto y desarrollar una aplicación para la visualización y control remoto de la misma.
- Analizar las variables ambientales dentro de la cámara de incubación, comparar con otros sistemas de control y verificar si existe una mejora en la producción de huevos.

CAPÍTULO II.

2 MARCO TEÓRICO.

2.1 ANTECEDENTES

La incubación artificial no es un proceso moderno, si bien los sistemas de control y monitoreo han ido avanzando con el paso de los años, esta se remonta a siglos de uso, Aristóteles menciona que los egipcios usaban estructuras similares a hornos para realizar el proceso de incubación, describiendo que usaban estiércol de camello para ofrecer la temperatura necesaria para la eclosión.

El primer intento de replicar el método de los egipcios se registra en 1588 por Jean Baptiste quien se vio forzado a dejar sus intentos debido a la inquisición española, en 1609 Cornelis Drebbel, inventó una incubadora que permitía mantener una temperatura constante, haciendo uso de un sistema de circulación de aire caliente alimentado por carbón, el aire caliente circulaba alrededor de una caja interna en donde estaban los huevos, con esta logro incubar algunos polluelos. En 1851 William James Cantelo, presentó su incubadora, la cual utilizaba un método que denominó “Hidro incubación”, con el cual según él ya había producido una gran variedad de aves, sin embargo, cualesquiera de estos métodos requerían la supervisión constante para mantener la estabilidad de la temperatura.

En el siglo XIX aparecen las primeras válvulas y mecanismos termostáticos automáticos siendo estos los que dieron paso a la incubación automática moderna, a partir de este punto se comienzan a registrar varias patentes, en las cuales el control de temperatura se encontraba totalmente automatizado [9].

Actualmente se pueden encontrar una gran variedad de incubadoras, ya sean manuales o totalmente automáticas, de uso casero o industrial, sin embargo cada una de estas necesita de la presencia continua de un operador que verifique el desarrollo apropiado de la incubación, en el trabajo “Diseño y construcción de un prototipo de incubadora con supervisión inteligente para la eclosión de huevos” [10] se implementa un modelo de incubadora que pretende automatizar el proceso de incubación tradicional, haciendo uso de un sistema de visión artificial, para reducir en gran medida la intervención hombre – máquina. De igual manera en el trabajo “An Automated Egg Incubator with Raspberry Pi-Based Camera Assisted Candling and R-CNN-based Maturity Detection” [11] es implementado un sistema de visión artificial basado en una red neuronal convolucional para detectar la condición y madurez de los huevos dentro de la incubadora.

2.2 INCUBACIÓN ARTIFICIAL

La incubación artificial es una forma que por medio de máquinas incubadoras permiten el desarrollo de las crías de aves, ofreciendo un medio ambiente adecuado y controlado.

Este proceso se ha utilizado desde los egipcios, los cuales introducían huevos en restos de estiércol y esperaban su eclosión, pero con el pasar los años se fueron implementando

sistemas complementarios que ayudaban de una manera más eficaz el proceso de incubación [10].

En el sector avícola existen dos técnicas que se aplican en la incubación como lo son:

2.2.1 MÉTODO DE ETAPA ÚNICA

Consiste en introducir una única carga de huevos en la incubadora, los cuales deben tener la misma edad embrionaria, una ventaja que presenta este método es que su sistema de control de temperatura, humedad, y nivel de CO₂ se mantiene en los valores predefinidos para el óptimo desarrollo de los huevos, por otro lado, este método requiere de una gran demanda energía y tiempo para poder cargar la incubadora por completo [12].

2.2.2 MÉTODO DE ETAPA MÚLTIPLE

Consiste en introducir varias cargas de huevos durante el periodo de incubación, lo que conlleva a tener huevos de diferentes edades embrionarias dentro de la incubadora, como principal desventaja no permite generar las condiciones óptimas para cada uno de los lotes de huevos, sin embargo, requiere de un menor consumo de energía ya que sus sistemas de control son básicos y los huevos con mayor tiempo de incubación se encargan de generar calor para los huevos con menor tiempo de incubación [13].

2.3 INCUBADORAS

Son dispositivos que reemplazan a la gallina en el proceso de incubación, capaces de replicar las condiciones de humedad, temperatura, y movimiento de una incubación natural, con una mayor precisión.

Estas se caracterizan principalmente por poseer sistemas que les permiten establecer un rango determinado en la temperatura y humedad necesaria para el tipo de huevo que se pretenda incubar, también poseen una mayor capacidad de carga siendo de cientos hasta miles de huevos por proceso de incubación

Hoy en día se pueden encontrar incubadoras ya sean simples para fines domésticos hasta modelos para producción industrial, entre las más comunes se pueden mencionar los siguientes tipos.

2.3.1 INCUBADORAS MANUALES.

Estas incubadoras permiten tener un mejor control sobre cada etapa de desarrollo del embrión. Lo que hace que esta actividad sea una experiencia única para el avicultor, además de ser un proceso más natural.



Figura 1. Incubadora manual Covatutto Mod.16 de 12 huevos.

Fuente: [14]

2.3.2 INCUBADORAS AUTOMÁTICAS

Se destacan por ser aparatos fáciles y seguros al momento de su uso, de igual forma tienen alta productividad debido a que se puede monitorear la temperatura, humedad, ventilación, y el sistema de volteo automático, haciendo la actividad más sencilla y asegurando que los polluelos nazcan sanos [15].



Figura 2. Incubadora automática HDD mini 56S para 55 huevos.

2.4 MANEJO DE LA INCUBACIÓN

El proceso de incubación debe llevarse de la manera más adecuada y de forma controlada para el óptimo desarrollo de los embriones, para lo cual antes de colocar los huevos en la cámara de incubación, estos deben pasar por un proceso de precalentamiento, con la finalidad de evitar un cambio brusco en la temperatura, existen dos métodos de realizar este “precalentamiento” como lo son:

- Aumentar la temperatura durante algunos periodos de tiempo en varias ocasiones durante su almacenamiento.
- Incrementar la temperatura durante algunas horas antes de ponerlos a incubar.

Antes de iniciar con el proceso de incubación los huevos deben estar debidamente seleccionados para tener un óptimo desarrollo, seguidamente se coloca la incubadora en un sitio que este grande, vacío y limpio apartado de fuentes de calor, verificar que no entre corrientes de aire para evitar cambios bruscos de la temperatura, y que el sol no pueda ingresar directamente al lugar donde se colocará la incubadora, se debe mantenerse una temperatura ambiental de 20 a 25°C, antes de ingresar los huevos se realiza una desinfección y secado de la cámara de incubación, después se enciende y calienta la incubadora, y finalmente se regula la temperatura y humedad para poder introducir los huevos.

Durante el día 1 al 17 de incubación se mantiene la humedad y temperatura en el rango establecido para el óptimo desarrollo del embrión, del día 18 hasta la eclosión se debe bajar la temperatura e incrementar la humedad en los valores recomendados, en el nacimiento de los pollitos se disminuye la humedad para que el pollo se seque, el tiempo máximo que el pollo debe permanecer en la incubadora luego de su nacimiento es de 24 horas [16].

2.5 FACTORES CLAVE DE LA INCUBACIÓN

Para que el desarrollo del huevo sea óptimo se debe mantener en constante monitoreo y control factores tales como: la temperatura, humedad, ventilación, y el movimiento del huevo dentro de la cámara de incubación.

2.5.1 LA TEMPERATURA

Es el factor más importante de todo el proceso de incubación, debido a que siempre se debe vigilar los cambios de temperatura, caso contrario afectará el desarrollo del ave. Los huevos en incubación son extremadamente sensibles a los cambios de temperatura. Además, este parámetro define la velocidad de desarrollo de los embriones dentro de los huevos. De manera general se recomienda que esté entre 37 y 38 °C, y disminuir esta entre 36 a 37 °C durante los 3 últimos días de la incubación (día 18 a 21). Por otro lado, se menciona que la temperatura de la incubadora y la que se da en la cáscara del huevo son totalmente diferentes oscilando entre 1 a 1.5°C [17].

El rango de temperatura varía dependiendo del tipo del ave que se vaya a incubar como se puede observar en la siguiente tabla:

Tabla 1. Temperatura Óptima de Incubación para Diferentes tipos de Aves.

Fuente: [17]

Tipo de Ave	Temperatura Óptima de Incubación
Gallina	37.7 °C
Pavo	37.2 °C
Pato	37.7 °C
Ganso	37.2 °C

2.5.2 LA HUMEDAD

Es otro factor importante en la incubación, pues, los huevos también son altamente susceptibles a cualquier cambio en este parámetro. Se debe hacer vigilancia constante de la humedad y corregir lo más rápido posible a través de la aspersión de agua. Ocasionalmente, se utilizan desinfectantes en esta agua para realizar una desinfección de los huevos cuando es oportuno. El huevo pierde hasta un 10 % de agua en el día 18 como parte del proceso de incubación. La humedad ambiental también tiene participación en este mecanismo.

Por lo tanto, es de gran importancia considerar algunos aspectos:

- El requerimiento de humedad del huevo se verá afectado por el peso, el tamaño y el color del mismo, siendo una humedad relativa (RH) de 50 a 55% óptima para huevos blanco y una RH de 55 a 60% para huevos de color café.
- En caso de una RH baja se puede compensar usando un humidificador o en tal caso una esponja húmeda.
- Además, cuando se esté por finalizar el proceso de incubación, la humedad se establecerá en un 75 a 80% para huevos blancos y en 80 a 85% para huevos cafés, para facilitar el resquebrajamiento del cascarón y facilitando el nacimiento de los polluelos [18].

Tabla 2. Rangos de Humedad, para diferentes tipos de aves.

Fuente: [17]

Tipo de Ave	Humedad Mínima %	Humedad Máxima %
Gallina	55	60
Pavo	55	60
Pato	53	58
Ganso	60	65

2.5.3 MOVIMIENTO

En condiciones naturales, las gallinas realizan la rotación de los huevos. En los procesos de incubación se debe asegurar la rotación de los huevos dependiendo el periodo de días en que esté la incubación. Esta rotación favorece el posicionamiento correcto del embrión dentro del huevo y evita que se pegue, ayudando a la formación de las membranas internas. Además, facilita la llegada de calor de manera uniforme a todos los huevos y en toda su estructura [17]

El volteo de los huevos es de carácter crítico durante la primera semana de desarrollo, esto debido a la distancia que existe entre la cáscara y el embrión, durante este periodo el embrión depende de la propagación de gases a través de su cáscara, para la obtención de O₂ y por consiguiente la eliminación de CO₂, además de prevenir la deshidratación y el desarrollo embrionario incorrecto [19].

Por otro lado, influye directamente en el porcentaje de incubabilidad del huevo, dado el caso de mantenerse entre 43° a 45° se obtiene un 91.41% de incubabilidad, mientras tanto al mantenerse en unos 15° o por debajo su incubabilidad es de un 50.91%.

Los factores tales como la frecuencia de volteo, el eje de colocación, el ángulo y el plano de rotación influyen en el desarrollo del polluelo, por tanto se ha comprobado que un mayor número de volteo al día y con mayor frecuencia genera un mayor porcentaje de incubabilidad, sin embargo esto influye de manera negativa operacionalmente, ya que se aumenta los costos en el mantenimiento del equipo que se encarga de este proceso, por lo que se toma una secuencia de volteo de 24 veces con una frecuencia de 1 hora, durante la primera etapa de incubación (18 días) [20].

2.5.4 VENTILACIÓN

La ventilación tiene como fin el refrescar o calentar el aire que rodea al huevo. Además, durante la incubación el huevo absorbe oxígeno y elimina anhídrido carbónico en gran cantidad, por lo que el continuo paso de aire es esencial en su desarrollo [21].

Para lo cual se usan dos formas de abastecer a la incubadora de aire fresco:

a) VENTILACIÓN ESTÁTICA

Esta ventilación se produce cuando el aire caliente que se acumula en la base de la incubadora se eleva generando un flujo de aire natural hacia la parte superior, siendo este método el más utilizado en incubadoras manuales por medio de orificios tanto en la parte superior como inferior, los cuales permiten el flujo de aire en el dispositivo [10].



Figura 3. Principio de Ventilación Estática.

b) VENTILACIÓN FORZADA

Este tipo de ventilación conlleva el uso de ventiladores para introducir aire fresco del exterior a la cámara de incubación y extraer el aire con CO₂ del interior, además nivela la temperatura y humedad, logrando la entrada de aire durante todo el proceso de incubación, sin la existencia de los ventiladores el intercambio de aire se obtendría por elevación y salida del aire caliente y la entrada de aire cerca de la base de la incubadora [16].

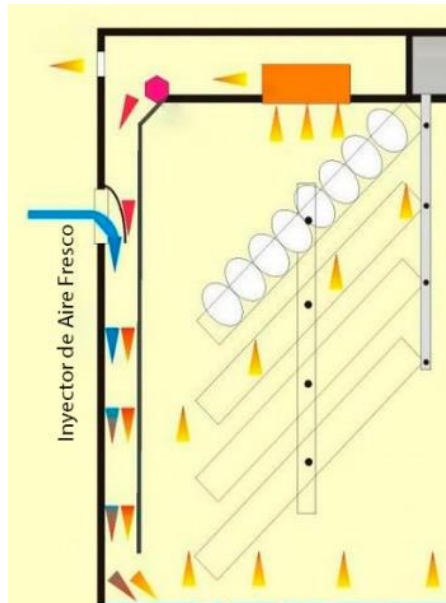


Figura 4. Principio de Ventilación Forzada.

Fuente: Pinterest

2.6 COMPONENTES DE UNA INCUBADORA

La incubadora debe replicar las condiciones naturales para el correcto desarrollo del huevo, por lo que mantener controlado el ambiente y el movimiento dentro de la cámara de incubación es esencial porque de ello depende la eficiencia del proceso [10].

2.6.1 ESTRUCTURA

Es una parte primordial, ya que en esta descansan las bandejas con los huevos, además de soportar las condiciones a las que serán sometidos estos para su desarrollo, por otro lado, el tamaño de la misma dependerá de la cantidad de huevos a incubar.

Ya sea básica o profesional, la incubadora debe disponer de una puerta para la manipulación de los huevos en caso de ser necesario y debe poseer una constante circulación de aire, casi en su totalidad poseen una ventana de vidrio cubierta que junto con un display ayudan a monitorear y controlar los procesos que realiza la incubadora [9].

Siendo los siguientes materiales los más usados para su fabricación:

a) ESTRUCTURA DE ALUMINIO

Es un material utilizado para realizar cualquier tipo de estructura ya que dispone de una resistencia y fuerza que ningún otro material lo tiene, además presenta un bajo peso que lo hace de fácil manejo y transporte, igualmente es tolerante ante el deterioro provocado por el entorno [22].



Figura 5. Incubadora Masalles Mod.2600-I HLC con estructura de aluminio.

Fuente:[23]

b) ESTRUCTURA DE MADERA

La madera es un material duro, aislante de la temperatura y sonido, es de fácil manipulación y que puede lograr dimensiones y formas necesarias, la conductividad térmica es baja por lo que la pérdida de calor es aceptable si el grosor de las paredes es de 19mm o más [24].



Figura 6. Incubadora de huevos con estructura de madera.

Fuente: Emilio Caceres, 2013

2.6.2 CONTROL DE TEMPERATURA Y HUMEDAD

Es uno de los sistemas principales que se encarga de monitorear y regular tanto la temperatura como la humedad dentro de la cámara de incubación, para lo cual se usan sensores electrónicos para la lectura de datos y una tarjeta de desarrollo que será la encargada de realizar las acciones que sean necesarias para mantener dichas condiciones dentro de los límites requeridos [10].



Figura 7. Controlador de temperatura y humedad.

Fuente: MercadoLibre

2.6.3 CONTROL DE MOVIMIENTO Y VENTILACIÓN

El sistema de control de movimiento será el encargado de controlar el ángulo de inclinación y de la frecuencia de volteo de los huevos dentro de la cámara de incubación, por otro lado, la ventilación se encarga de proporcionar una adecuada oxigenación a toda la cámara de incubación evitando la proliferación de gases que puedan desprender los huevos y conservando un ambiente óptimo para su desarrollo [10].

2.7 IOT

El internet de las cosas o más conocido como IoT, es una de las tecnologías que más se nombran en la actualidad, y es considerado como una revolución en el mundo de las comunicaciones, al cual se puede definir como una red inteligente que permite el intercambio de información y comunicación entre dispositivos inteligentes [25], por otro lado, se lo define como una plataforma que es capaz de conectar dispositivos para el monitoreo de aspectos cotidianos [26].

2.7.1 ARQUITECTURA DE IOT

Un inconveniente que presenta el IoT es que su concepto es demasiado amplio, por lo cual no cuenta con una arquitectura uniforme, por lo tanto, se plantean las siguientes arquitecturas:

- **Arquitectura UIT.** -De acuerdo con las recomendaciones de la Unión Internacional de Telecomunicaciones (UIT), la arquitectura de IoT consiste en: Capa de Acceso, Capa de Red, Capa Inter-Lógica y Capa Aplicación.
- **Arquitectura de 3 capas.** - La arquitectura IoT hace uso de la representación de un modelamiento en capas y se divide básicamente en 3 con funciones independientes: Capa Física, Capa de Virtualización y Capa de Aplicación [26].
- **Arquitectura estándar.** - Típicamente la arquitectura de IoT es interpretada según las organizaciones o los proveedores de servicios, sin embargo, la arquitectura básica de IoT se puede representar con un modelo de cuatro capas como lo son: Capa de Infraestructura, Capa Gestión de servicios. Capa de Aplicación, Capa de Negocios/Capa Empresarial [27].

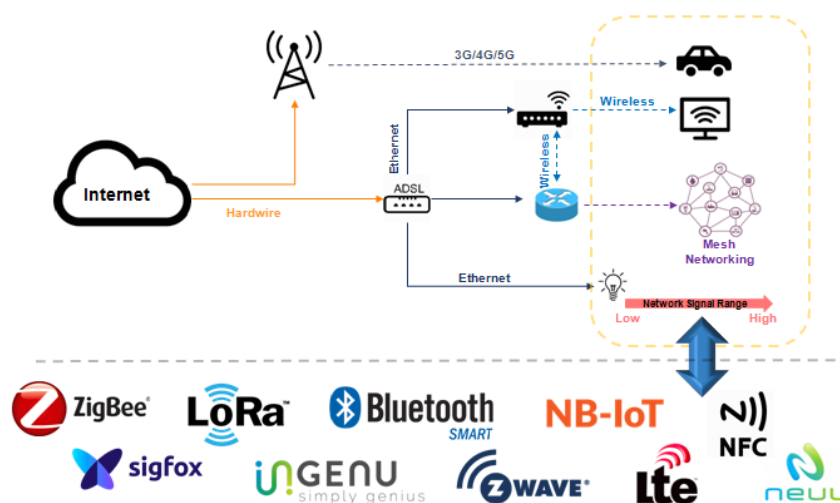


Figura 8. Arquitectura de una Red IoT y protocolos Disponibles.

Fuente:[28]

2.7.2 PROTOCOLOS DE IOT

Sea cual sea el modelo de referencia que se tome para su representación la arquitectura de IoT en cada una de sus capas puede llegar a utilizar una diversidad de protocolos, como lo pueden ser:

- **Capa de Infraestructura.** - en esta capa se engloban las capas, física, red y transporte de datos por lo cual se implementan protocolos como: Ethernet, Bluetooth, Zigbee, LoRaWan, NFC y WiMax, para la capa física, por otro lado, para la capa de red se usan protocolos como: IPv4, IPv6, 6LoWPAN, IPv6 sobre Bluetooth, por

último, en la capa de transporte se implementan: TCP, UDP, TLS, TSMP entre otros [27].

- **Capa Gestión de Servicios.** - esta capa diferencia las redes IoT, ya que estos dispositivos IoT necesitan encontrar otros dispositivos, por lo cual se implementan protocolos de gestión y descubrimiento de servicios como: DNS -SD, mDNS, UPnP, HiperCat, Bluetooth Beacons, Wi-Fi Aware y Shazam.
- **Capa Aplicación.** - IoT implementa protocolos como MQTT, CoAP, XMPP, SMCP, SMQTT, DDS, HTTP/2, JavaScript IoT o WebSocket en su capa de aplicación, a continuación, se definen algunos de ellos [27].

MQTT (MQ Telemetry Transport) es una implementación ligera y sencilla de un protocolo de mensajería, su diseño se orienta a redes con bajo ancho de banda y altos niveles de latencia.

CoAP (protocolo de aplicación restringida) es un protocolo de transferencia especializado para el uso con nodos y redes de baja potencia y de altos niveles de pérdidas, este está diseñado para tener fácil interacción con HTTP y su operación se basa en el protocolo UDP de capa transporte del modelo TCP/IP [26].

XMPP (protocolo de presencia y mensajería extensible) es un protocolo de mensajería basado en XML, que evolucionó para extender a HTML, permitiendo estructurar datos y escalar fácilmente las redes IoT, siendo este usado principalmente para aplicaciones en tiempo real [29].

- **Capa Empresarial.** – esta capa se encuentra al final de IoT y no forma parte de la red de comunicación puede distinguirse por ser una nube o un servidor que brinde soluciones empresariales o industriales al sistema IoT, o puede definirse como la interconexión de dispositivos para brindar ERP (solución de planificación de recursos empresariales) [27].

2.8 VISIÓN ARTIFICIAL

La visión artificial se la define como la ciencia de programar un computador para procesar imágenes o videos para llegar incluso a entenderlos. También se la presenta como la transformación de datos desde un fotograma o video, los cuales pueden llegar a representar una decisión o incluso ser una nueva representación.

Los procesos o etapas que se llevan a cabo en la visión artificial son:

- **Adquisición de Datos (Imagen).** –Se captura y se digitaliza la imagen.
- **Procesamiento.** –Se preprocesa la imagen en la cual se eliminan partes de esta, las cuales no tienen relevancia para la aplicación.
- **Detección de bordes.** –Se separan los objetos de relevancia y se descarta el fondo de la imagen.

- Segmentación. - se selecciona los píxeles en función de los colores primarios RGB, o por un modelo basado en los componentes de Matiz, Saturación y Valor (HSV), para representar algunos colores.
- Extracción de características. - Se definen las características que se deben de tomar en cuenta en la imagen.
- Reconocimiento y localización. - Se identifican los objetos que cumplan con las características ya definidas y se puede restringir el área de la imagen a interpretar.
- Interpretación. - Se da un sentido a la imagen ya procesada como puede ser el reconocimiento de un rostro, de un color o de la forma de un objeto.

Cada uno de los procesos no tienen dependencia entre sí por lo tanto la implementación de cada uno de ellos depende específicamente de la aplicación que se vaya a implementar [26].

CAPÍTULO III

3 METODOLOGÍA

3.1 TIPO DE INVESTIGACIÓN

3.1.1 INVESTIGACIÓN NO EXPERIMENTAL

Este tipo de investigación se realiza sin manipular las variables, es decir se realiza la observación de los fenómenos que le sucedan al objeto de estudio.

3.2 DISEÑO DE INVESTIGACIÓN

3.2.1 MÉTODOS DE INVESTIGACIÓN

El método de investigación que se utilizará forma parte de los métodos lógicos, siendo este de inducción completa debido a que las conclusiones se obtendrán del estudio de todos los elementos que forman el objeto de investigación, por lo cual se debe recopilar datos de otros sistemas de incubación y compararlos con el prototipo propuesto.

3.3 TÉCNICAS DE RECOLECCIÓN DE DATOS

La investigación no experimental presenta dos tipos: De manera transversal y longitudinal, cada uno de ellos son utilizados por el investigador con intención de estudio diferente. Para el presente proyecto se aplicará la investigación longitudinal la cual realiza la recolección de los datos a través de distintos periodos del tiempo con el fin de realizar inferencias sobre la evolución, causa, consecuencias y la relación que tienen estas variables.

Técnica de observación: Este tipo de técnica es utilizada para la recolección de los datos mediante la observación del comportamiento del objeto de estudio de igual manera se realiza un registro ordenado para obtener resultados óptimos, esto se lo hace sin necesidad de intervenir directamente con el objeto de estudio.

3.4 POBLACIÓN DE ESTUDIO Y TAMAÑO DE MUESTRA

3.4.1 POBLACIÓN

En el presente proyecto se pretende evaluar 2 procesos de incubación donde en el primero se utilizará un sistema de incubación existente y en el segundo se implementará el sistema propuesto, en ambos se analizará las variables ambientales dentro de la cámara de incubación, partiendo de valores fijados para el óptimo desarrollo de los embriones.

3.4.2 MUESTRA

Se tomará una muestra no aleatoria que consiste en los datos de un proceso de incubación (alrededor de 21 días).

3.5 HIPÓTESIS

El diseño e implementación del sistema de monitoreo y control ambiental basado en IoT optimiza la producción de huevos.

3.5.1 OPERACIONALIZACIÓN DE VARIABLES

a) Variable Independiente

Control ambiental: temperatura, humedad; monitoreo: adquisición de imágenes y datos.

b) Variable Dependiente

Optimización del proceso de incubación.

Tabla 3. Análisis de las variables dependiente e independientes

VARIABLE	CONCEPTO	INDICADORES	INSTRUMENTO
Control ambiental: temperatura, humedad, monitoreo: adquisición de imágenes y datos	Capacidad para mantener la temperatura y humedad entre los valores óptimos. Información que muestra la situación actual de incubación.	<ul style="list-style-type: none">- Nivel de humedad, y nivel de temperatura.- Latencia.	<ul style="list-style-type: none">- Software de desarrollo Arduino y Python.- Dispositivos de control IoT. (PC, Dispositivos móviles).
Optimización del proceso de incubación.	Información que verifica si el sistema cumple con la optimización del proceso de incubación.	<ul style="list-style-type: none">- Porcentaje de producción.	<ul style="list-style-type: none">- Base de datos y- Excel.

3.6 MÉTODO DE ANÁLISIS

Se utilizará la estadística descriptiva para realizar el análisis e interpretación de los resultados de la presente investigación, con el fin de recolectar y analizar los datos obtenidos en el proceso de incubación, mediante tablas y gráficos indicando las diferenciaciones entre un sistema ya existente y el propuesto.

3.7 PROCESAMIENTO DE DATOS

Atreves de los criterios obtenidos del marco teórico, objetivos y el enfoque de la investigación se realiza un análisis exhaustivo para generar un prototipo que cumpla con las características necesarias para manejar la incubación de huevos de gallina y aumentar la producción, además de generar una base de datos durante cada proceso de incubación para mantener un registro más efectivo de los parámetros establecidos.

Por lo cual el desarrollo del proyecto se llevó a cabo como se muestra en el flujograma de la Figura 9.

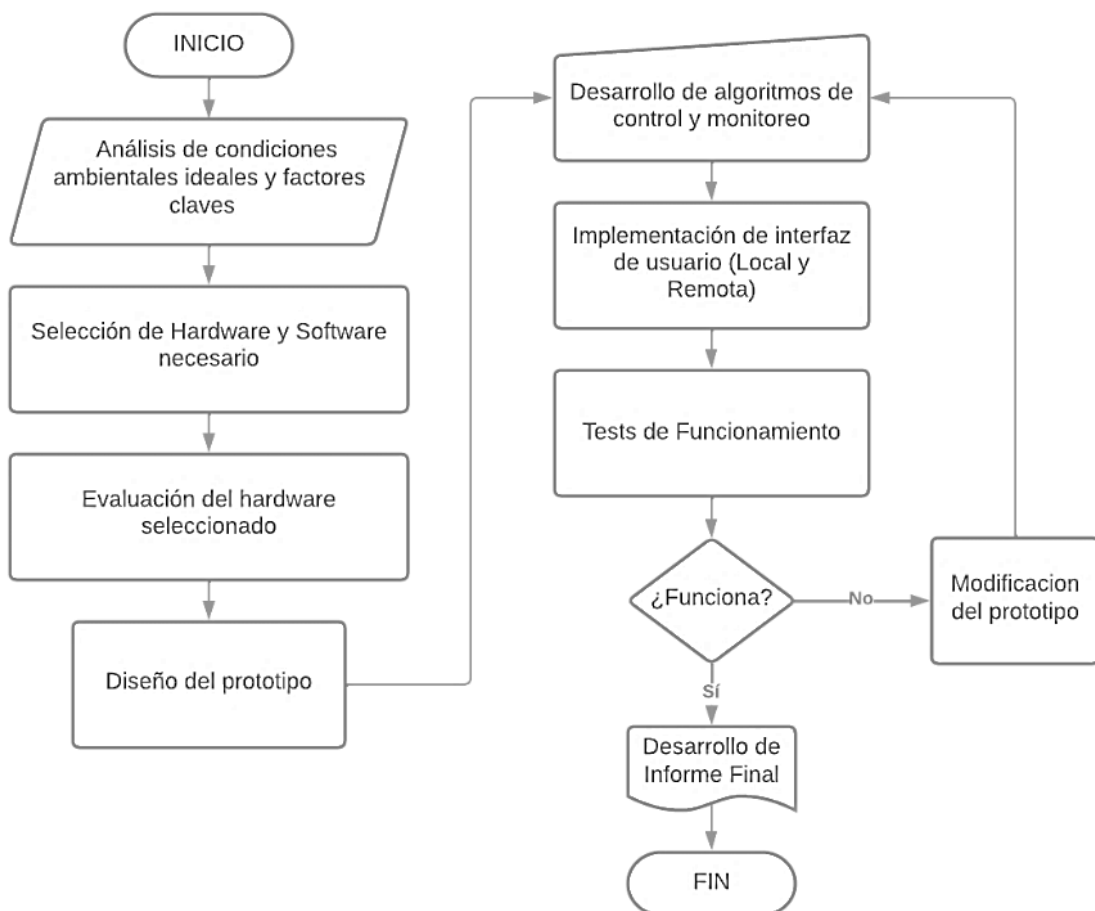


Figura 9. Flujograma del desarrollo del proyecto

- Analizar los principales parámetros y factores claves que influyen en el proceso de incubación, procurando la obtención de información pertinente para el desarrollo del proyecto
- Determinar el hardware y software a utilizar para el correcto procesamiento de las señales y la evaluación de los huevos que se encuentren aptos para continuar en la incubación.
- Evaluar las características del hardware seleccionado para la implementación del prototipo y distinguir sus respectivas ventajas y desventajas.

- Diseñar el prototipo del sistema de control y monitoreo automatizado basado en IoT.
- Desarrollar los algoritmos necesarios para el control ambiental y el manejo, visualización, almacenamiento característico de un sistema IoT.
- Implementar la interfaz gráfica local y remota para el monitoreo y control del proceso de incubación.
- Realizar pruebas de funcionamiento del prototipo del sistema de incubación, verificando el cumplimiento de los parámetros requeridos y propuestos en los objetivos.
- Elaborar el informe final del proyecto de investigación una vez que se haya llevado a cabo la implementación del prototipo, incluyendo toda la información recopilada y producida durante el desarrollo del mismo.

3.7.1 ANÁLISIS DE FACTIBILIDAD

El presente proyecto de investigación es factible tanto técnico, bibliográfica y económicamente:

Es factible técnicamente debido a que los equipos y dispositivos electrónicos y mecánicos necesarios se encuentra disponible en el país y su implementación se basa en software y hardware libre, bibliográficamente puesto que la información se obtuvo de publicaciones científicas, repositorios universitarios, tesis e internet, y económicamente ya que su financiamiento lo cubre el propio investigador.

3.8 REQUERIMIENTOS DEL PROTOTIPO

Para realizar el proyecto, se requirió que el sistema fuese modular y estable, considerando las particularidades que surgen durante el proceso de incubación y de la morfología de los huevos de gallina, siendo el enfoque de éste, generar condiciones ambientales favorables para el desarrollo de los huevos y mantener un constante monitoreo de cada uno de ellos.

Para mantener las condiciones ambientales idóneas se implementa técnicas de control feedback como controladores PID, o control on/off, usando un microcontrolador que se encarga de su adquisición y procesamiento; Por otro lado, para el monitoreo se emplean técnicas de visión artificial y adquisición de imágenes, siendo procesadas y almacenadas por una microcomputadora.

3.8.1 SELECCIÓN DE HARDWARE

Para la implementación del sistema se procedió a realizar la selección de los dispositivos y componentes que cumplan con las características necesarias por lo cual se analizó los modelos disponibles en el mercado actual.

Para el control y monitoreo se seleccionaron dos tarjetas de desarrollo como lo son: un microcontrolador Arduino Mega 2560 y una microcomputadora Raspberry Pi 4B, las cuales presentan las características necesarias para la implantación del sistema.



Figura 10. Dispositivos embebidos, Arduino Mega 2560 y Raspberry Pi 4

Tabla 4. Características de los Dispositivos Embebidos

Placa	Arduino Mega	Raspberry Pi 4B
Procesador	N/A	Broadcom BCM2711, Quad core Cortex-A72
Microcontrolador	Atmega2560	N/A
Pines E/S	54 (15- PWM),16 (analógicos)	40 pines GPIO
RAM	8KB SRAM	1 a 4GB SDRAM
Bluetooth/WiFi/Ethernet	N/A	5.0/2.4 GHz/Gigabit
Voltaje de Funcionamiento	5v DC	5v DC por conector USB-C

Para la obtención de las variables ambientales se seleccionó el sensor AHT10, el cual tiene unas dimensiones reducidas y permite obtener de manera precisa la temperatura y humedad, por otro lado, se seleccionó una cámara Logitech-B525 para la obtención de imágenes para el respectivo monitoreo.

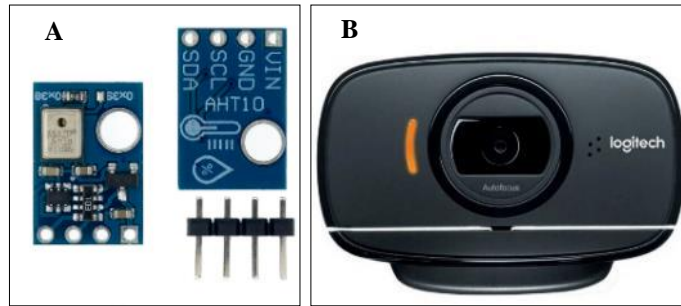


Figura 11. A) Sensor AHT10. B) Cámara Logitech B525

Tabla 5. Características Sensor AHT10.

Modelo		AHT10	
Alimentación		3.3V	
Señal de salida		I2C	
Temperatura		Humedad	
Rango	-40°C a 85°C	Rango	De 0% RH a 100% RH
Precisión	±0.3°C	Precisión	± 2°C
Resolucion	0.01°C	Resolucion	0.024°C

También, se utilizó como actuadores una niquelina de estufa eléctrica como fuente de calor, una niquelina eléctrica de agua para generar la humedad apropiada, un servo motor DS3235 para el giro de los huevos, motores paso a paso bipolares NEMA 17 para el movimiento de la cámara a través de la incubadora, y ventiladores para generar un flujo constante de aire.



Figura 12. A) Niquelina Eléctrica, B) Motor NEMA 17, C) Servo motor DS3235 D) Ventilador PWM

Tabla 6. Características Servomotor y Motor Bipolar

Características	Servomotor DS3235	Motor bipolar NEMA17
Torque máximo	35 kg.cm	130Mn.m
Velocidad máxima	0.13 seg/60	-
Voltaje de operación	5v-7.4v	2.8V-12V
Tamaño	40.1 x 20 x 38.1 mm	42x42x23mm
Peso	60g	280g
Corriente nominal	-	1.5 A
Angulo de paso	-	1.8
Rango de temperatura de trabajo	0c-55c	-20C a 50C

3.8.2 DISEÑO DEL PROTOTIPO

En el presente punto se describe el diseño implementado para el sistema propuesto, para ello se tuvo en cuenta los requerimientos necesarios para ser un sistema de incubación artificial óptimo.

Para lo cual se dividió el proceso de diseño en cuatro puntos fundamentales como lo son los siguientes:

a) DISEÑO ESTRUCTURAL

Para el diseño de la estructura se seleccionó la madera como materia prima, por ser uno de los materiales más adecuados para mantener el ambiente bajo las condiciones requeridas, para el diseño se usó el software de modelado 3D Cinema4D.

La estructura se divide en tres partes como lo son: la cámara de incubación, en donde se desarrolla la incubación, una base móvil similar a una cubeta tradicional pero modificada para los requerimientos del sistema, y una estructura fija para el desplazamiento de la cámara.

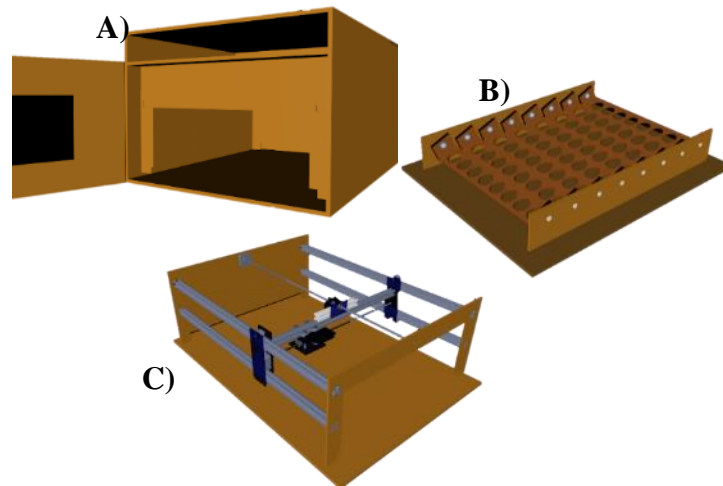


Figura 13. A) Cámara de incubación. B) Cubeta para los Huevos. C) Estructura de Desplazamiento

b) DISEÑO DEL SISTEMA DE CONTROL DE TEMPERATURA-HUMEDAD

Para el control de temperatura se diseñó un control de bucle cerrado PID en la siguiente figura se muestra la estructura que cumple el sistema de control.

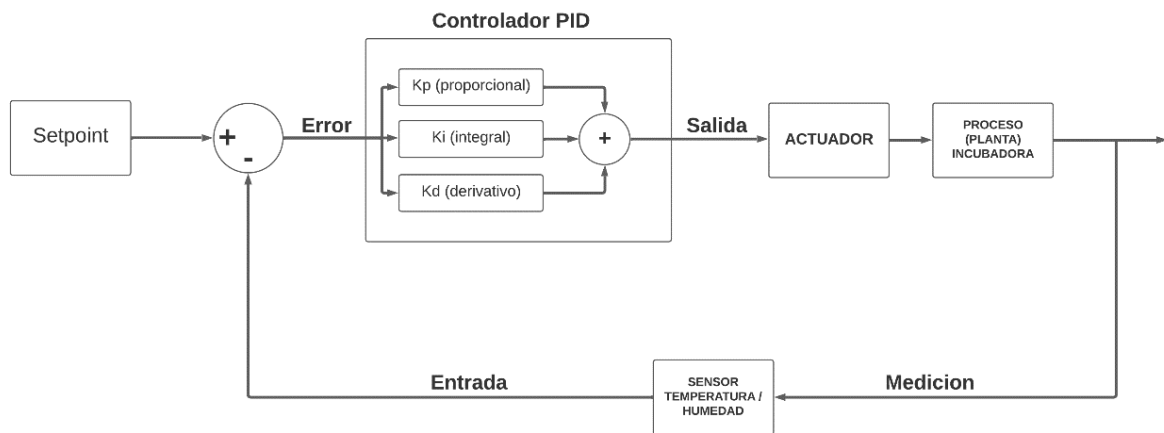


Figura 14. Diagrama general del Controlador PID a implementar

Para ello primero se realiza una caracterización del sistema a lazo abierto a través de su respuesta a un cambio escalón, obteniendo el siguiente comportamiento:

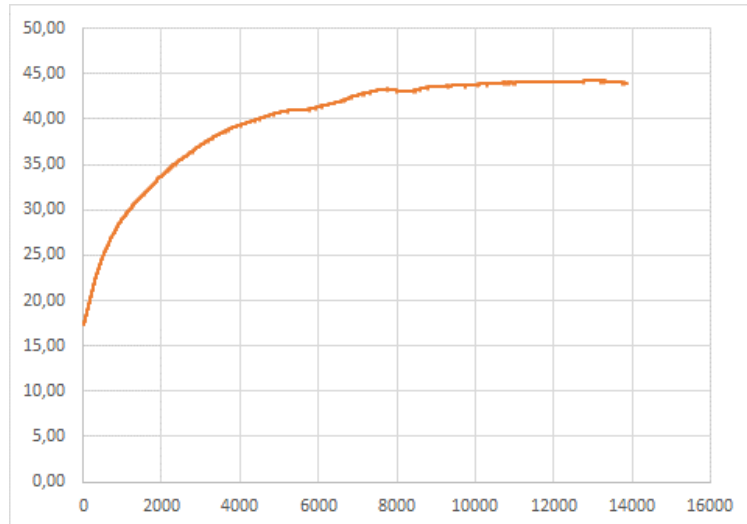


Figura 15. Curva de comportamiento del sistema a un Cambio escalón.

Como se puede apreciar en la figura la respuesta del sistema es de primer orden por lo cual el sistema se representa con la siguiente fórmula:

$$G(s) = \frac{K}{\tau s + 1}$$

Para la obtención de las variables K y τs se realizan operaciones apoyándose de los datos recolectados, para ello se necesita que estos valores se encuentren en variables de desviación obteniendo la siguiente respuesta:

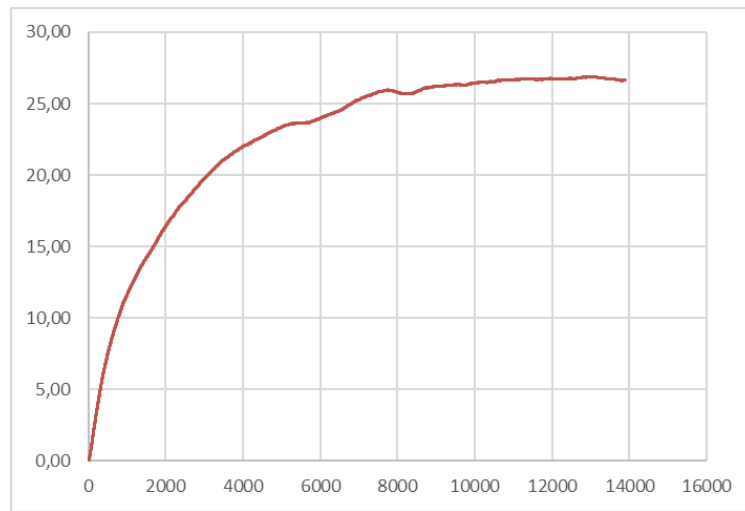
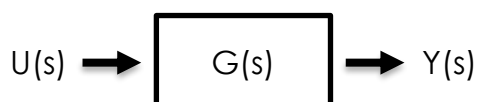


Figura 16. Curva de comportamiento del sistema en variables de Desviación

Ahora el sistema queda representado como se muestra a continuación:



$U(s)$: que es el cambio escalón de 0 a 100,

$G(s)$: es el sistema de control a laso abierto, y

$Y(s)$: es la respuesta del sistema al escalón.

Por lo tanto, la respuesta queda representada de la siguiente manera:

$$Y(s) = \frac{K}{\tau s + 1} U(s)$$

Siendo su representación en el tiempo:

$$y(t) = K\Delta u(1 - e^{-t/\tau})$$

Para obtener los valore de K y τ los obtenemos sabiendo que:

$$\Delta u = 100$$

$$K = \frac{\text{valor final}}{\Delta u} = \frac{26.64}{100} = 0.2664$$

Para obtener τ buscaremos el valor de $y(t)$ cuando el tiempo es igual a τ entonces:

$$y(t) = 0.632(K\Delta u) = 0.632(0.2664 * 100) = 16.83$$

ahora este valor lo buscamos en los datos obtenidos del sistema y τ será igual al tiempo en el que se encuentre ese valor:

2095	100	34,17	16,81
2096	100	34,19	16,81
2097	100	34,19	16,81
2098	100	34,20	16,82
2099	100	34,21	16,83
2100	100	34,21	16,83
2101	100	34,22	16,84
2102	100	34,22	16,84
2103	100	34,22	16,84
2104	100	34,23	16,85

Figura 17. Selección de τ con respecto a los datos obtenidos del sistema

En este caso para este sistema $\tau = 2100$

ahora se puede representar el sistema con los valores obtenidos aplicando la formula $y(t)$ a cada uno de los valores de nuestro sistema, obteniendo la síguete respuesta:

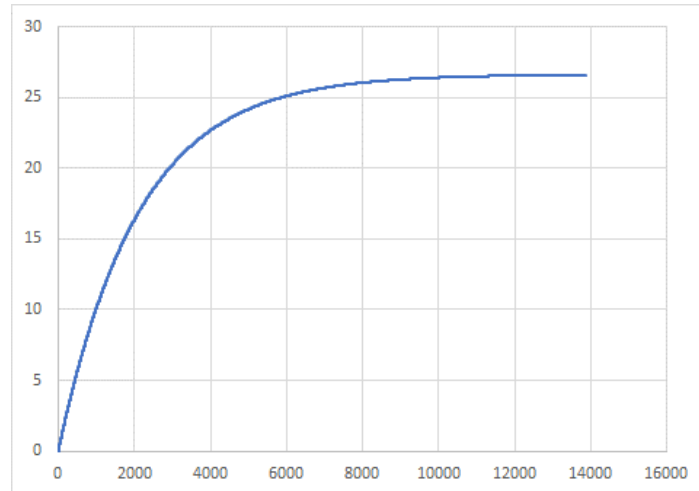


Figura 18. Respuesta de $y(t)$ al sistema.

Una vez representada la respuesta con los valores obtenidos se aprecia que ésta es similar a la respuesta que presenta nuestro sistema y procedemos a implementar el controlador PID para un sistema de primer orden.

Para lo cual el sistema se somete a simulaciones para obtener los valores determinantes que ayuden a mantener un control adecuado de la temperatura, con ayuda del Software Matlab y obteniendo los siguientes resultados:

Tabla 7. Simulación de valores para K_p y T_i del Controlador PID

N° Simulación	Valor Proporcional (K_p)	Valor integral (T_i)	Resultado
1	1	100	Presenta mayor número de oscilaciones sobrepasando el setpoint por 8 °C y su tiempo de reacción y estabilización es muy lento
2	1	300	Presenta oscilaciones sobrepasando el setpoint por 4 °C y su tiempo de reacción y estabilización es muy lento
3	1	800	Presenta un sobrepaso del setpoint por 0.6 °C, su tiempo de reacción y estabilización es muy lento

4	1	1800	No presenta ninguna oscilación y llega al setpoint, su tiempo de reacción y estabilización es mucho más lento que los anteriores
5	2.5	800	Presenta un sobrepaso del setpoint de 2°C, su tiempo de reacción y estabilización es normal
6	5	1800	Presenta un sobrepaso del setpoint por 1°C, su tiempo de reacción y estabilización es normal, no llega al setpoint
7	6.4	1800	Presenta un sobrepaso del setpoint de 2°C, su tiempo de reacción y estabilización es normal, no llega al setpoint
8	7	1000	Presenta un sobrepaso del setpoint de 1.7°C, su tiempo de reacción y estabilización es normal, no llega al setpoint
9	9.2	1800	Presenta un sobrepaso del setpoint de 0.2°C, su tiempo de reacción y estabilización es adecuado
10	20	1000	Presenta un sobrepaso del setpoint de 1.9°C, su tiempo de reacción y estabilización es rápido, no llega al setpoint
11	35	1800	Sin sobrepaso ni oscilaciones, tiempo de reacción y estabilización muy rápido, no llega al setpoint
12	40	300	Presenta un sobrepaso del setpoint de 4.3°C, su tiempo de reacción y estabilización es muy rápido, no llega al setpoint

Por lo cual se tomó los valores de Kp 9.2 y Ti 1800, para su uso en el código de control del microcontrolador.

Por otro lado, para el control de humedad se usa un control de encendido y apagado ya que el umbral de humedad que requiere la incubación permite el uso de este tipo de control.

c) DISEÑO DEL SISTEMA DE MONITOREO

El sistema encargado de mover la cámara digital la cual captura las imágenes está compuesto por motores y tornillos que se encargan de desplazar la cámara sobre la cubeta de huevos. De esta manera, se obtienen las imágenes necesarias para llevar a cabo el monitoreo de cada huevo.

Para la implementación se toma en cuenta diseños ya existentes de dispositivos CNC, e impresoras 3d, sin embargo, el enfoque del sistema de monitoreo es desplazar la cámara a través de la cubeta por lo cual se prescindirá del eje z, haciendo uso de un sistema de 2 dimensiones.

Considerando las bases, se procede al desarrollo del sistema mecánico para el desplazamiento requerido tomando como referencia el sistema tradicional de coordenadas cartesianas.

El sistema de coordenadas cartesianas utiliza ejes ortogonales X, Y que se mueven de forma lineal y permiten el posicionamiento preciso de la herramienta. En la Figura 19 se muestra una vista del diseño mecánico.

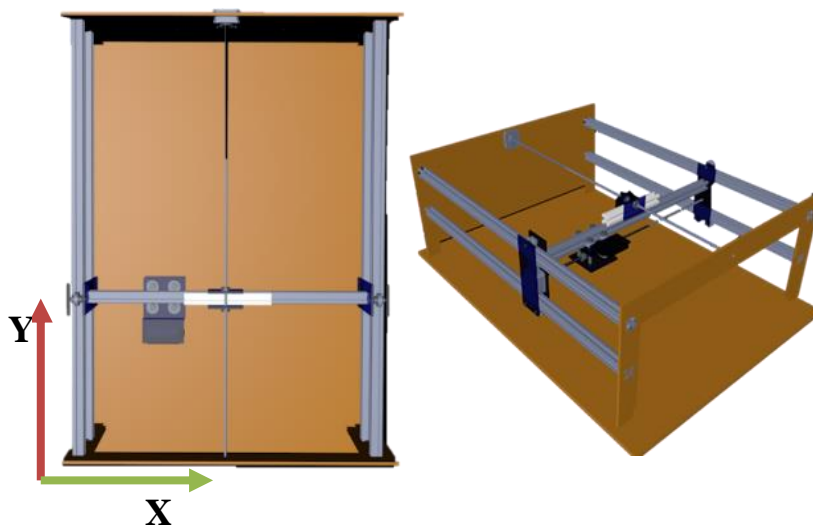


Figura 19. Modelado en 3D del sistema mecánico a implementar.

Para el control de movimiento del mecanismo mecánico se utilizó el lenguaje de programación C++ de arduino para desarrollar una función dentro del script de arduino para el control de los motores.

La conexión de los motores y sus drivers al microcontrolador se muestran en la siguiente figura.

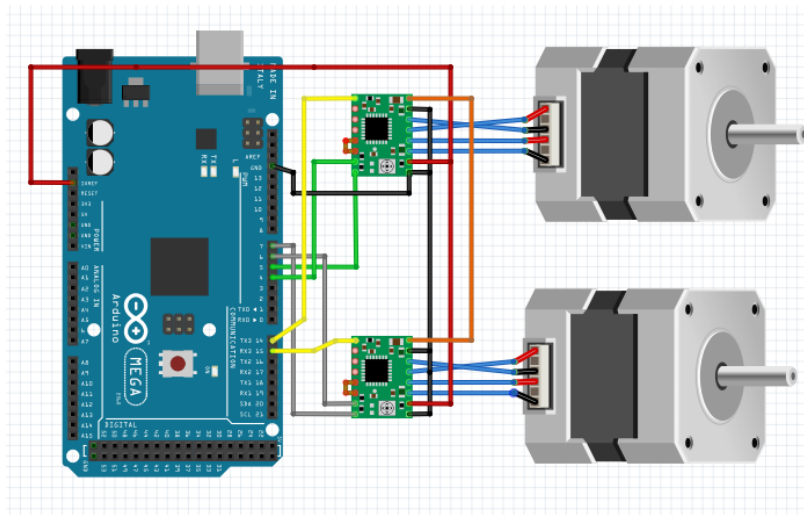


Figura 20. Diagrama de conexiones para el mecanismo de movimiento.

Para el procesamiento de las imágenes obtenidas se utiliza el lenguaje de programación Python que es manejado por la Raspberry Pi, usando la herramienta de procesamiento OpenCV, para ello se debe de instalar la librería en la raspberry, una vez instalada se procede con la creación de un script que capture las imágenes y dependiendo de las condiciones se realicen las siguientes operaciones:

- Identificación de Huevos fértiles a través de la comparación en diferentes días del proceso.

Para este proceso se realiza la captura al trasluz de cada sujeto de prueba por tres días durante el día 7 ,14 y 18 para así determinar si el huevo se encuentra desarrollándose o no, el algoritmo para este proceso se muestra en la Figura 21.

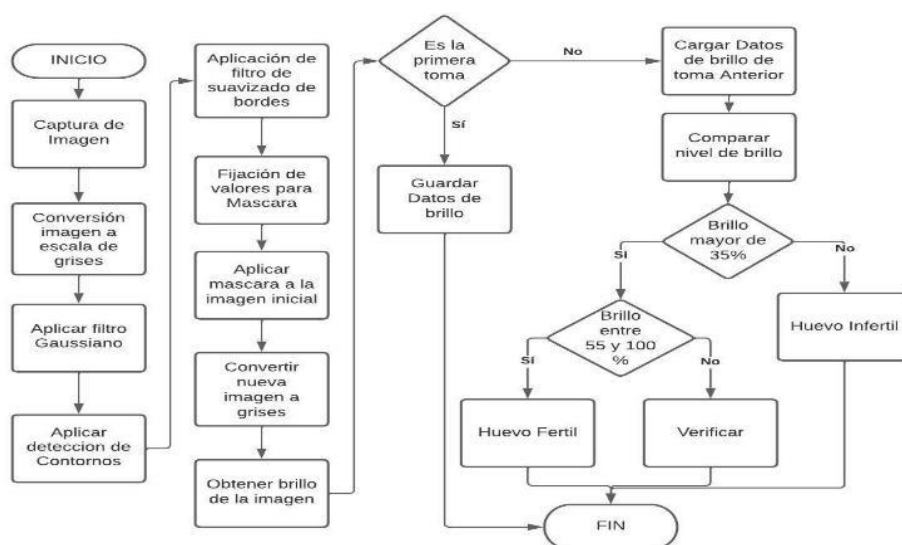


Figura 21. Algoritmo para el Proceso de Ovoscopia.

- Identificación de anomalías en los huevos y detección de huevos a punto de eclosionar.

Para este proceso se realiza capturas de cada sujeto con iluminación directa dos veces en el día para la detección de cualquier anomalía superficial que pudiera presentarse durante la incubación, en la siguiente Figura 22, se presenta el algoritmo para este proceso.

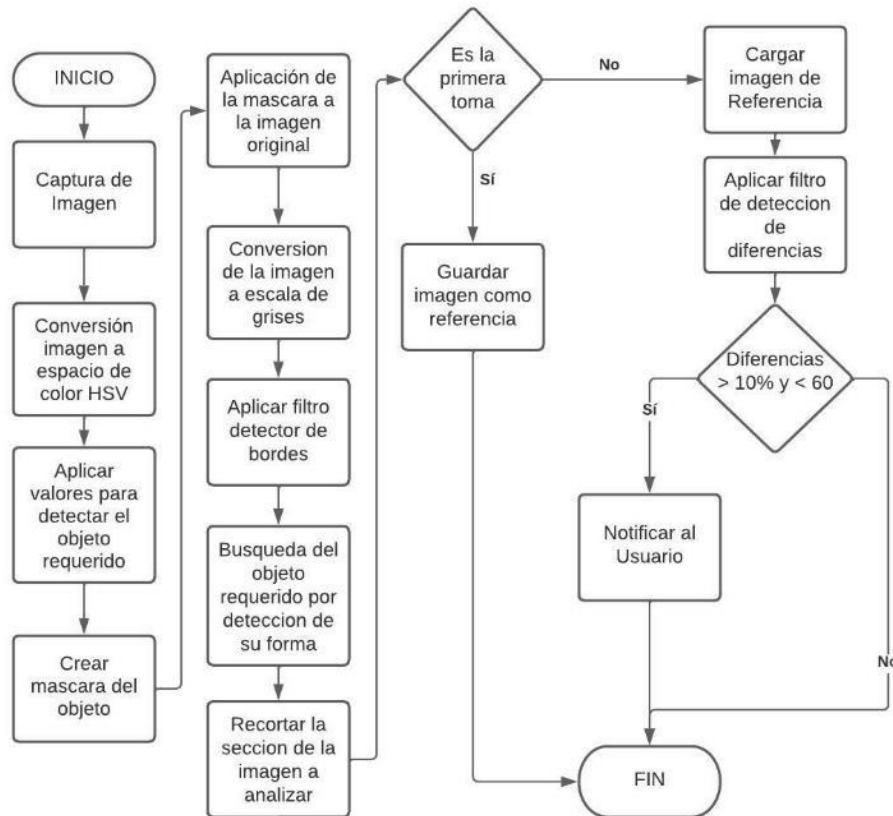


Figura 22. Algoritmo aplicado para la Detección de anomalías en los huevos.

d) SISTEMA PARA LA COMUNICACIÓN IOT

Para el diseño del sistema propuesto se utiliza un sistema IoT, basado en un dispositivo, un Gateway, una nube, y una aplicación de usuario, en la Figura 23, se puede observar la constitución de cada uno de estos.

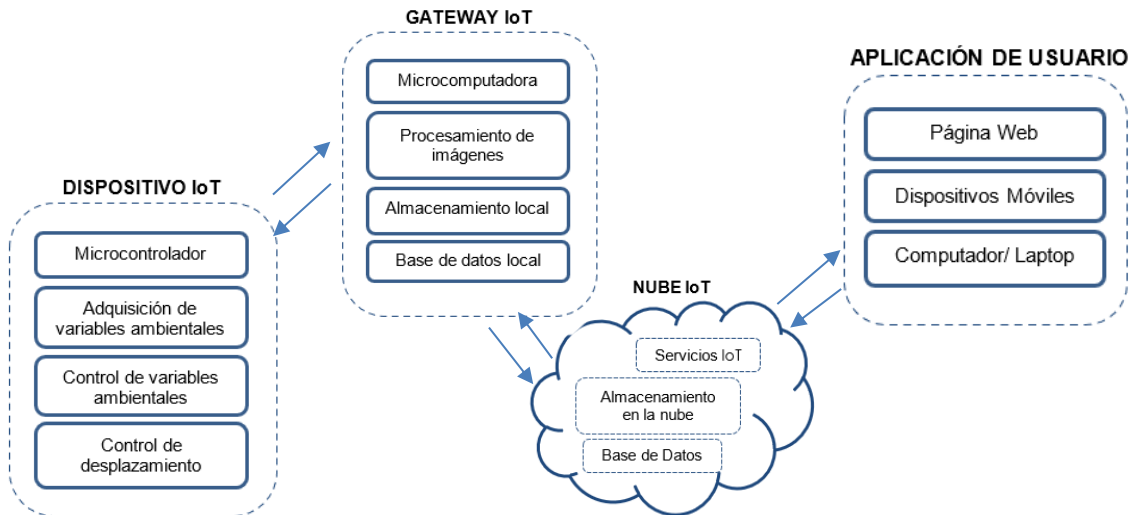


Figura 23. Diagrama de constitución de los dispositivos en el sistema IoT a implementar.

- Dispositivo IoT. este constara de un dispositivo embebido el cual permite obtener los datos de sensores y actuadores.
- Gateway IoT. Principalmente se utilizan dispositivos que tengan la capacidad de conectarse a internet actuando como intermediario entre dispositivo IoT y la nube.
- Nube IoT. Se utilizan servicios de almacenamiento tales como Firebase, AWS, Microsoft Azure, o Brokers MQTT para el almacenamiento y procesamiento de datos.
- Aplicaciones de usuario. Se hace uso de aplicaciones móviles o aplicaciones web para el control y configuración del sistema.

Para el almacenamiento de la información obtenida se utiliza una base de datos tanto local como en la nube, localmente se utiliza una base de datos SQLite, ya que los datos que se almacenan no constituyen una carga significativa, por otro lado, para el almacenamiento en la nube se utiliza Firebase de Google, siendo este un modelo NoSQL que al igual que SQLite permite un almacenamiento organizado de la información.

Para el control y visualización en tiempo real a través de la aplicación de usuario se utiliza un bróker MQTT y Firebase Realtime, el Bróker MQTT se encarga de enviar las imágenes y las señales de control de las funcionalidades implementadas en el sistema, por otra parte, Firebase Realtime se encarga de enviar los valores obtenidos de los sensores de temperatura y humedad hacia la aplicación de usuario.

e) DISEÑO DE INTERFAZ GRÁFICA LOCAL Y REMOTA

Para el desarrollo de la aplicación local se empleó la herramienta Kivy que es una librería o módulo de interfaz gráfica bajo el lenguaje de programación Python cuyo entorno se debe descargar y configurar para raspberry, se creó una interfaz con 7 pantallas, ya que esta es mostrada a través de un display colocado en el prototipo, por lo tanto, se debe de poder

acceder a cada funcionalidad que proporciona el sistema, a continuación, se describen cada una de ellas.

- **Usuario**, en esta se visualiza los diferentes usuarios que se han llegado a vincular al dispositivo, también permite gestionar a cada uno de ellos.
- **Gráficos**, aquí podemos graficar los datos obtenidos durante cada día de las variables de humedad y temperatura.

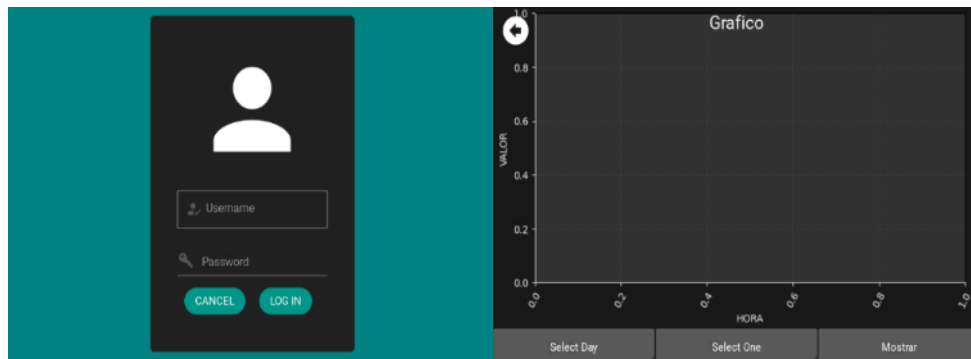


Figura 24. Pantalla en Kivy de Usuario y Gráficos

- **Cámara**, en esta pantalla podemos hacer uso de la cámara incorporada dentro del dispositivo, además de brindar los controles para el desplazamiento, el almacenamiento de imágenes y el control sobre la bandeja de incubación.
- **Registro**, muestra de manera detallada un registro del comportamiento de las variables ambientales de todo el proceso de incubación.

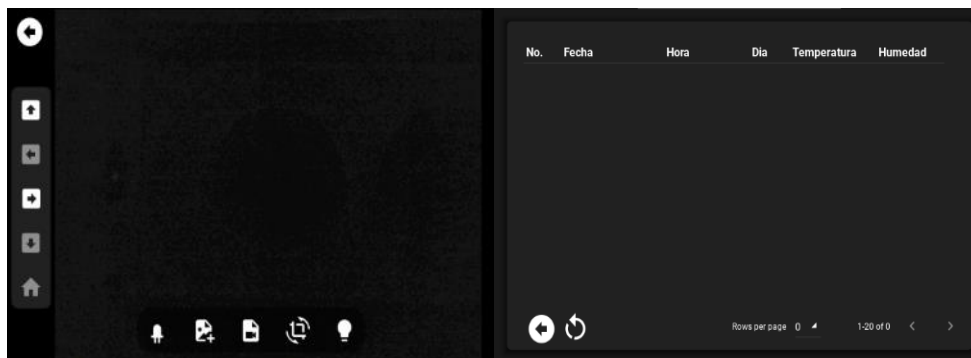


Figura 25. Pantalla en Kivy de Cámara y Registro.

- **Notificaciones**, en esta sección se muestran las alarmas, los cambios que han ocurrido en la configuración inicial y cada acción que hayan cambiado durante la incubación.
- **Configuración**, se muestran tanto las configuraciones del sistema, de incubación y de control del prototipo.

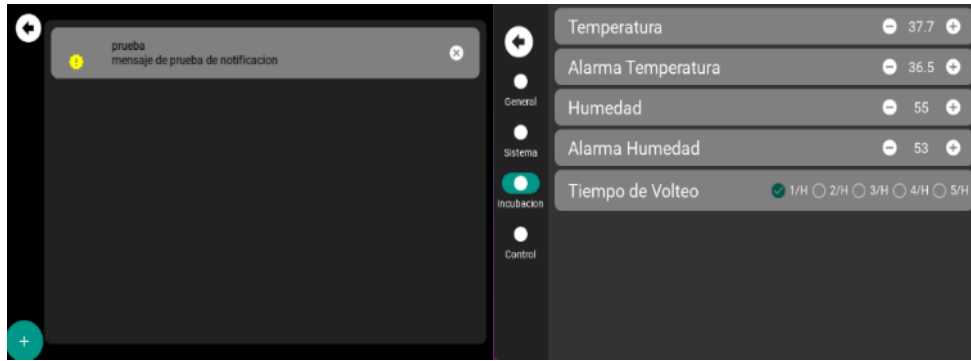


Figura 26. Pantalla en Kivy de Notificaciones y Configuración

Además de estas pantallas existe una pantalla principal en la cual se muestran los datos de temperatura y humedad, así como los valores máximos y mínimos de cada variable, también muestra un reloj con la hora actual y un contador de días.

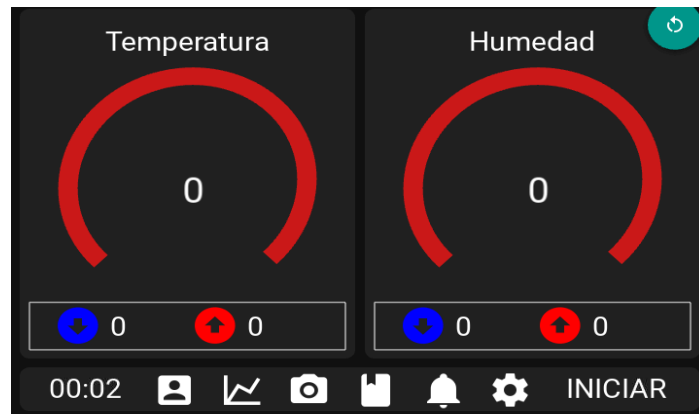


Figura 27. Pantalla de Visualización principal.

La interfaz del sistema se encuentra configurada de tal manera que mientras no se inicie con la incubación ningún dato se verá almacenado y se podrá navegar por cada una de las pantallas, sin necesidad de iniciar un proceso de incubación.

Para el desarrollo de la aplicación remota se utilizó el lenguaje de programación Dart del Framework Flutter para desarrollar una aplicación que pueda ser multiplataforma, para este proyecto se llevó a cabo la implementación de una interfaz para Android.

La aplicación Android consta de una pantalla principal en la cual el usuario se registra o ingresa a su cuenta ya sea con una cuenta de Google o por correo, al momento de garantizar el ingreso a su cuenta se muestra un pantalla con una barra de navegación en la cual se puede acceder a las diferentes funcionalidades al igual que en la aplicación local, tales como , la visualización de las variables ambientales en tiempo real y su respectiva graficación, el control y acceso a la cámara y a su sistema de desplazamiento, un registro en el cual se puede acceder a la base de datos almacenados en la nube, también un historial de las alarmas y

cambios del sistema, y por último el apartado de configuración en el que de igual manera se pueden cambiar la configuración del dispositivo.

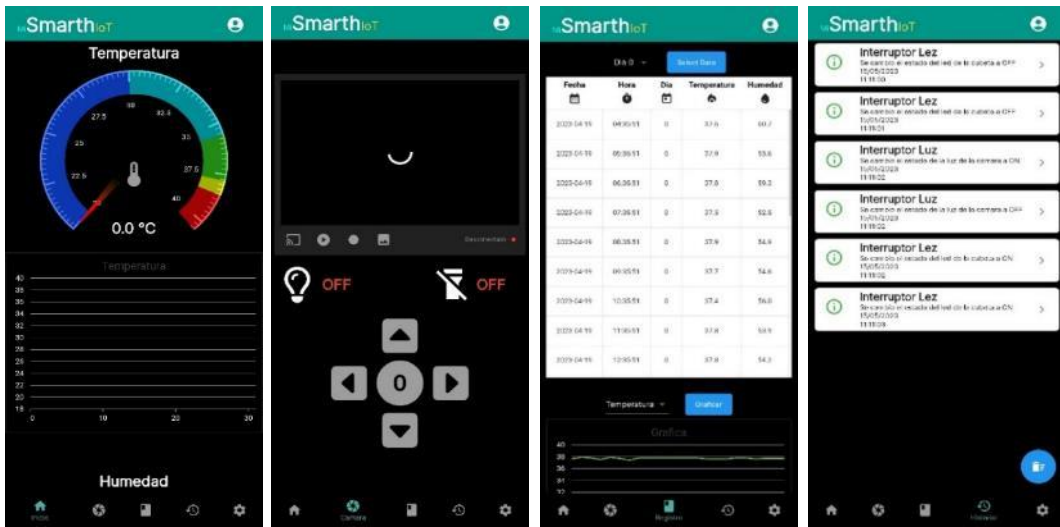


Figura 28. Interfaz Gráfica Remota (Dispositivo Android).

3.9 IMPLEMENTACIÓN DEL PROTOTIPO

Para la implementación se utilizaron los materiales y dispositivos descritos en el capítulo anterior, además, la disposición y ubicación de los dispositivos se tomaron como referencia a partir de las recomendaciones de otros investigadores que previamente trabajaron en proyectos similares. Después de realizar el diseño tridimensional de la estructura del prototipo, se llevó a cabo su construcción de acuerdo a dicho diseño.



Figura 29. Armado de la Cámara de incubación, Cubeta para los Huevos y Estructura de Desplazamiento.

Una vez realizado el armado de las diferentes estructuras se procedió a realizar el cableado y circuito del prototipo, para realizar las pruebas de funcionamiento de cada uno de ellos. En la figura se muestra las pruebas realizadas al sistema encargado de la ovoscopia.



Figura 30. Prueba de funcionamiento del sistema de otoscopia.

También se procedió a realizar pruebas para determinar el ángulo de giro del servomotor para obtener un ángulo apropiado para el giro de los huevos, en la siguiente figura se muestra el sistema usado.



Figura 31. Sistema de movimientos de Huevos.

Para la determinación del ángulo de 45° y -45° de los huevos se procedió aplicando diversos ángulos al servo motor en la Tabla 8 se muestran los resultados de las pruebas.

Tabla 8. Parámetros para el giro de las Regletas de Huevos.

Pruebas Sistema de Giro			
N° prueba	Ángulo Servomotor	Ángulo Regletas de huevos	Observaciones
1	80	0	El servomotor se mueve por momentos, no afecta al ángulo de las regletas
2	60	22.2	El ángulo de las regletas no llega a 45°
3	54	40.2	El ángulo de las regletas se aproxima a 45°
4	90	-16	El ángulo de las regletas no llega a -45°
5	106	-39.1	El ángulo de las regletas no llega a -45°
6	110	-42.4	El ángulo de las regletas se aproxima -45°

Una vez armado el sistema de desplazamiento se procede a realizar la respectiva prueba de funcionamiento en la cual se determinó que el giro horario de los motores desplaza al eje Y hacia adelante y al eje X hacia la derecha y el sentido antihorario los desplaza hacia atrás y hacia la izquierda respectivamente.



Figura 32. Sistema de Desplazamiento de la cámara.

También se realizaron pruebas para determinar los parámetros adecuados para cada motor en la Tabla 9 se muestra las pruebas hechas en el Eje Y, y en la Tabla 10 se muestran las realizadas en el Eje X.

Tabla 9. Parámetros para el movimiento del sistema mecánico encargado de desplazar la cámara a lo largo de la bandeja de huevos - Eje Y.

EJE Y			
N° Prueba	Velocidad de Giro	Tiempo Medido	Observaciones
1	100	55.82 s	Desplazamiento lento, vibraciones indeseadas, posicionamiento correcto de cámara
2	250	38.83	Desplazamiento lento, vibraciones indeseadas, posicionamiento correcto de cámara
3	500	21.09	Desplazamiento lento, posicionamiento correcto de cámara
4	750	12.64	Desplazamiento normal, posicionamiento correcto de cámara
5	1000	9.28	Desplazamiento rápido, posicionamiento correcto de cámara
6	1250	8.13	Desplazamiento irregular, posición incorrecta de cámara
7	2500	5.58	Atasco de eje, posición incorrecta de cámara

Tabla 10. Parámetros para el movimiento del sistema mecánico encargado de desplazar la cámara a lo largo de la bandeja de huevos - Eje X.

EJE X			
N° Prueba	Velocidad de Giro	Tiempo Medido	Observaciones
1	100	59.45 s	Desplazamiento lento, vibraciones indeseadas, posicionamiento correcto de cámara

2	250	24.48	Desplazamiento lento, vibraciones indeseadas, posicionamiento correcto de cámara
3	500	14.15	Desplazamiento lento, posicionamiento correcto de cámara
4	750	8.50	Desplazamiento normal, posicionamiento correcto de cámara
5	1000	6.62	Desplazamiento rápido, posicionamiento correcto de cámara
6	1250	4.78	Atasco de eje, posición incorrecta de cámara
7	2500	2.96	Atasco de eje, posición incorrecta de cámara

Por último, se realizó la conexión de cada uno de los sistemas implementados, A continuación, en la figura se muestra su conexión.

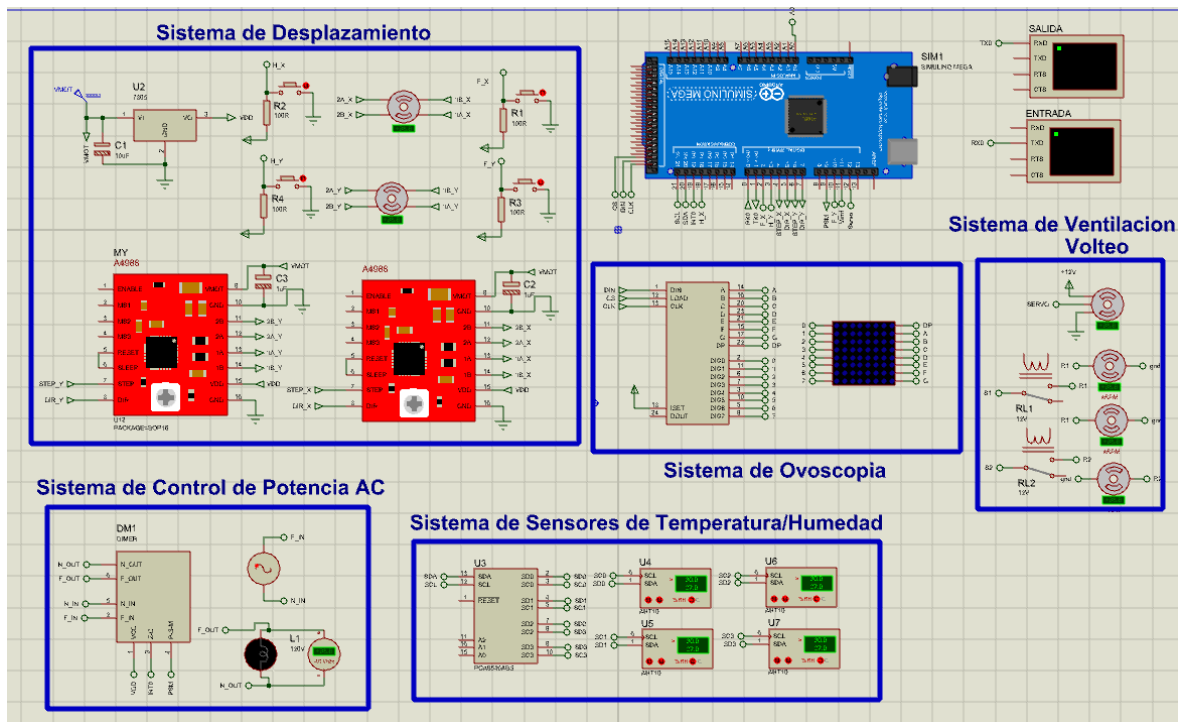


Figura 33. Conexión de los sistemas implementados en el proyecto (Simulador Proteus)

CAPÍTULO IV.

4 RESULTADOS Y DISCUSIÓN

Para evaluar el desempeño del sistema implementado es necesario realizar pruebas, las cuales permitirán obtener datos que ayuden a determinar un óptimo funcionamiento, tomando como prioridad las variables ambientales ya que de estas dependen una buena natalidad dentro de la incubación.

Por lo cual en el siguiente capítulo se detalla los datos obtenidos de dos procesos de incubación, siendo el primer proceso realizado en una incubadora existente en el mercado actual, y el segundo en el prototipo propuesto, siendo cada proceso de 21 días completos sin interrupciones.

4.1 PRIMER PROCESO DE INCUBACIÓN

4.1.1 SELECCIÓN DE HUEVOS FÉRTILES

Para la selección de huevos fértiles se llevó a cabo una búsqueda en empresas especializadas en la producción e incubación de aves de corral para seleccionar aquellos que cumplan con los criterios de calidad requeridos, siendo estos adquiridos de la empresa Avícola Avigofri, ubicada en la provincia del Carchi; los cuales se muestran a continuación en la Figura 34.



Figura 34. Colocación de los huevos fértiles en la incubadora.

Cabe resaltar que en la primera prueba se utilizó una incubadora automática HDD mini 56S con una capacidad máxima de 55 huevos.

4.1.2 INICIO DEL PROCESO DE INCUBACIÓN

Una vez colocados los huevos seleccionados en la incubadora se procede a activarla y a configurar los parámetros necesarios tales como temperatura máxima, humedad y sus respectivas alarmas.



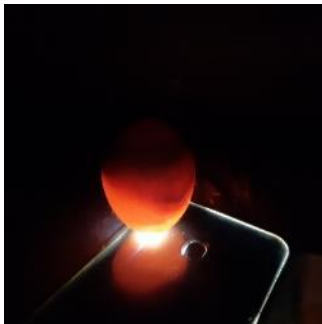



Figura 35. Encendido de la incubadora e inicio del Proceso de Incubación.

4.1.3 PRUEBAS DE OVOSCOPIA

Una vez iniciado el proceso de incubación se realizó pruebas de ovoscopia para lo cual la incubadora contaba con su propio sistema, sin embargo, este sistema de ovoscopia por su construcción no permitió visualizar de buena manera si los huevos se estaban desarrollando por lo cual se optó por realizar las pruebas manualmente. Resumiéndose en la siguiente tabla.

Tabla 11. Resultados de la Ovoscopia realizada en los días 9,14 y 18.

Días de Incubación	Detalles	Huevo fértil	Huevo infértil
9 días	Se puede realizar este proceso a partir de los primeros siete días de haber iniciada la incubación, para poder desechar los huevos infértiles.		
14 días	con esta segunda prueba se puede observar un mayor desarrollo en el embrión del huevo, notándose en éste claramente las arterias.		

<p>18 días</p>	<p>En esta última prueba se puede observar un área del huevo totalmente cubierta, lo cual indica el posicionamiento del polluelo para su nacimiento.</p>		
-----------------------	--	--	---

4.1.4 FIN DE LA INCUBACIÓN

Luego de haber transcurrido los 21 días, llega el momento de la eclosión de los huevos, ésta puede darse desde el día 19, sin embargo, para este proceso no se obtuvo ningún nacimiento, por lo cual se esperó un día más por posibles retardos, sin ninguna muestra de un posible nacimiento. A continuación, se muestran aquellos huevos que no lograron eclosionar.



Figura 36. Resultados. Mortalidad en etapa inicial y media del proceso de incubación.

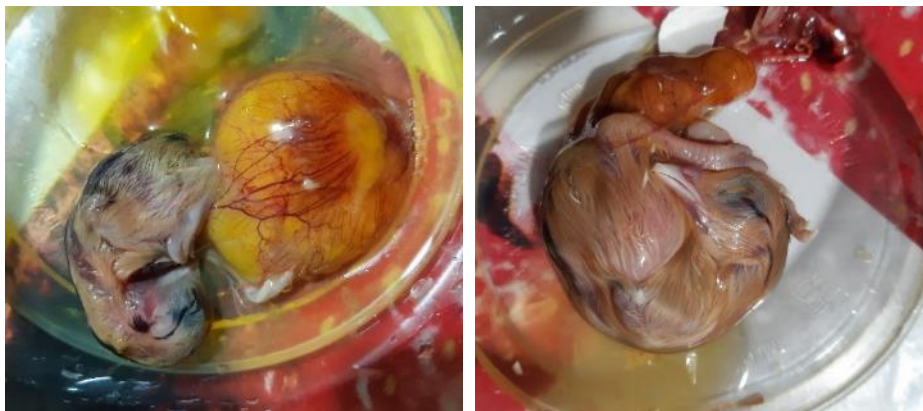


Figura 37. Resultados, Mortalidad en etapa final del proceso de incubación.

4.1.5 ANÁLISIS DE LOS PARÁMETROS DE HUMEDAD Y TEMPERATURA

Para el análisis de los parámetros ambientales se tomó como referencia el promedio de las mediciones de un sensor de temperatura y humedad externo y a los propios sensores de la incubadora ya que en pruebas de funcionamiento se notó un déficit en la medición de estos parámetros por parte de los sensores propios. A continuación, se presentan los datos obtenidos del proceso de incubación de forma resumida.

Tabla 12. Datos de temperatura y humedad obtenidos durante la incubación.

Día	Temperatura Promedio	Humedad Promedio	Dato -SetPoint Temperatura	Dato- SetPoint Humedad
0	37,80	56,47	0,10	-3,53
1	37,78	52,47	0,08	-7,53
2	37,75	51,16	0,05	-8,84
3	37,72	49,44	0,02	-10,56
4	37,74	51,22	0,04	-8,78
5	37,71	48,84	0,01	-11,16
6	37,64	50,24	-0,06	-9,76
7	37,65	49,33	-0,05	-10,67
8	37,73	51,81	0,03	-8,19
9	37,68	52,35	-0,02	-7,65
10	37,69	55,39	-0,01	-4,61
11	37,77	52,23	0,07	-7,77
12	37,73	50,27	0,03	-9,73
13	37,74	51,78	0,04	-8,22
14	37,70	49,96	0,00	-10,04
15	37,70	50,26	0,00	-9,74
16	37,73	53,01	0,03	-6,99
17	37,72	50,03	0,02	-9,97
18	37,64	49,58	-0,06	-10,42
19	37,58	54,68	1,08	-10,32
20	36,88	55,99	0,38	-9,01
21	36,50	62,85	0,00	-2,15

Una vez obtenidos los datos del periodo de incubación, se realizó una comparación de los datos de temperatura y humedad con los setpoints establecidos como se muestra en la Tabla 12. Cabe mencionar que se dividió el periodo en dos etapas: los primeros 18 días con un setpoint de 37.7°C de temperatura y 60% de humedad, y los últimos 3 días con un setpoint de 36.5°C de temperatura y 65% de humedad.

Con el fin de evaluar la capacidad del sistema para mantener las condiciones establecidas, se realizaron análisis estadísticos utilizando medidas como la media, la desviación estándar y la varianza. Estas medidas proporcionan información sobre la tendencia central, la

dispersión y la variabilidad de los datos, permitiéndonos comprender mejor el desempeño del sistema en relación a los setpoints establecidos.

Tabla 13. Valores característicos para el análisis de la temperatura.

Temperatura		
Día 0 al 18		
Media	Desviación Estándar	Varianza
0.04 °C	0.027°C	0.0007
Día 19 al 21		
Media	Desviación Estándar	Varianza
0.5°C	0.548°C	0.3

En los resultados que se muestran en la Tabla 13, se puede observar que durante los primeros 18 días, la temperatura promedio se mantuvo muy cercana a 37.7°C, ya que los valores de la desviación estándar y media se acercan a cero, indicando que el sistema fue capaz de mantener la temperatura estable y con poca variabilidad en este periodo.

Por otro lado, en los últimos 3 días con un setpoint de 36.5°C, se registró valores que muestran una mayor variabilidad y una desviación más significativa del valor establecido.

Tabla 14. Valores característicos para el análisis de la humedad.

Humedad		
Día 0 al 18		
Media	Desviación Estándar	Varianza
8.6%	2.016%	4.066
Día 19 al 21		
Media	Desviación Estándar	Varianza
7.2%	4.387%	19.254

En cuanto a la humedad sus datos se muestran en la Tabla 14, en los cuales se aprecia que durante los primeros 18 días su variabilidad es relativamente alta y su media expresa una tendencia a mantenerse por debajo de su valor óptimo, de igual manera en los últimos 3 días mantiene el comportamiento inicial, con la diferencia que su variabilidad presenta un valor considerablemente más alto.

A continuación, se muestran las gráficas del comportamiento de temperatura y humedad durante el proceso de incubación.

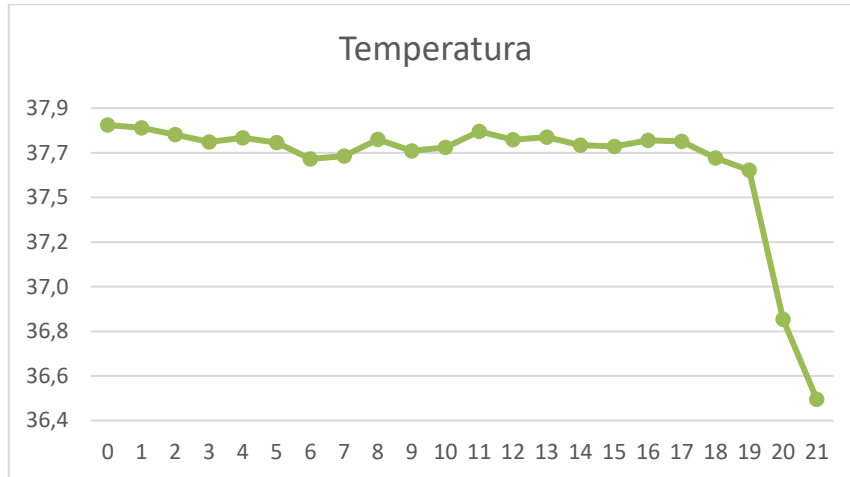


Figura 38. Gráfica de Temperatura del proceso de incubacion.

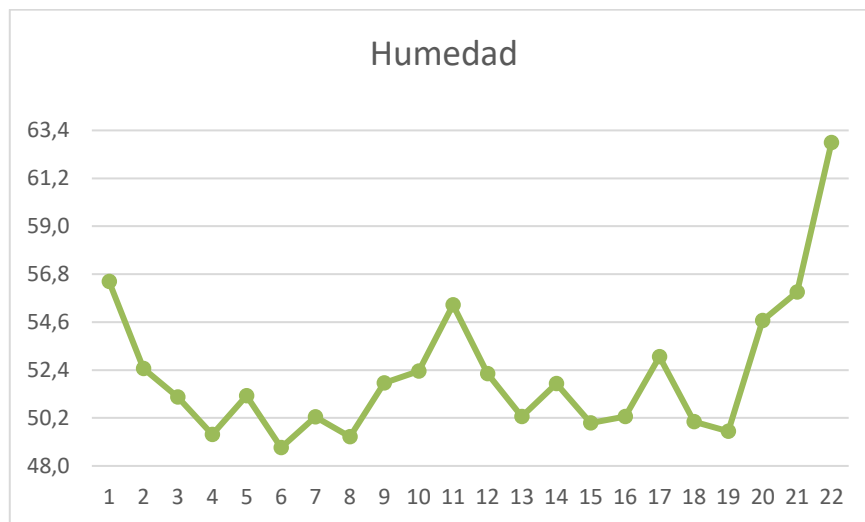


Figura 39. Gráfica de Humedad del proceso de Incubación.

Finalmente se observó que, el sistema de incubación existente, registró un error promedio de 0.26 °C en la temperatura. Sin embargo, se debe considerar que el sensor incorporado en dicho sistema presentaba una falta de precisión, lo que resultaba en mediciones diferentes a las obtenidas mediante sensores externos. Para mejorar la capacidad de incubación, fue necesario realizar ajustes en la configuración de temperatura y asegurar una mayor precisión en las lecturas de temperatura. Asimismo, el sensor de humedad también mostraba imprecisiones, reflejando un error promedio de 7.9% además este carecía del control necesario para su ajuste.

4.1.6 ANÁLISIS DE RESULTADOS

Se procede al análisis de los resultados obtenidos después de haber concluido con el primer proceso de incubación mostrando los mismos en la Tabla 15.

Tabla 15. Resultados del rendimiento en la producción de la incubadora.

Total, de huevos Incubados	Huevos Infértiles	Huevos Fértiles	Mortalidad de Huevos	Natalidad de Huevos
55	24	31	31	0

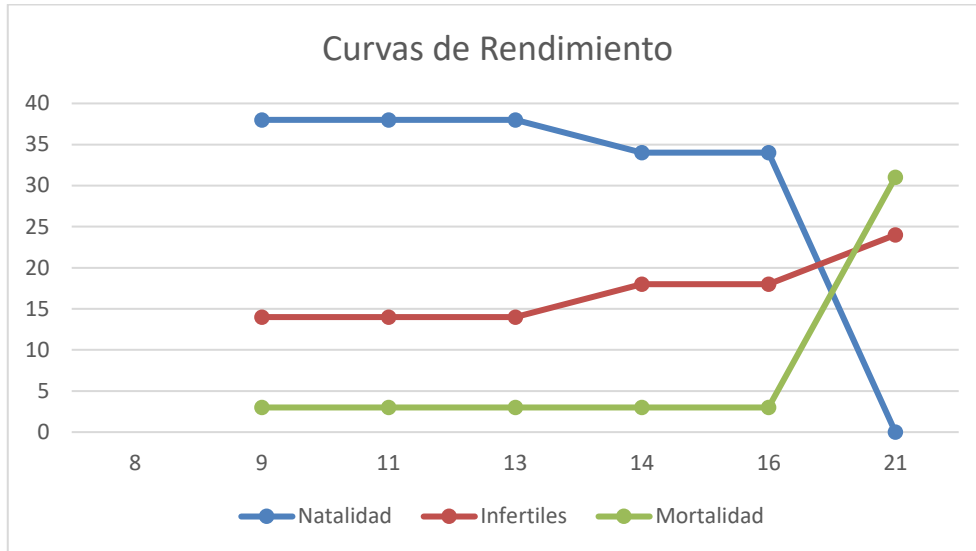


Figura 40. Curva de rendimiento del sistema

Los resultados obtenidos en el sistema de incubación adquirido fueron desfavorables, como se evidencia en la Figura 40 y Figura 41. Ninguno de los huevos fértiles logró eclosionar, lo que resultó en una muerte prematura de los polluelos. Esto indica una eficiencia de incubación del 0%, ya que no se registraron eclosiones exitosas. Estos hallazgos resaltan la necesidad de mejorar las condiciones de incubación para garantizar una mayor tasa de eclosión y mejorar la eficiencia reproductiva en futuros procesos.

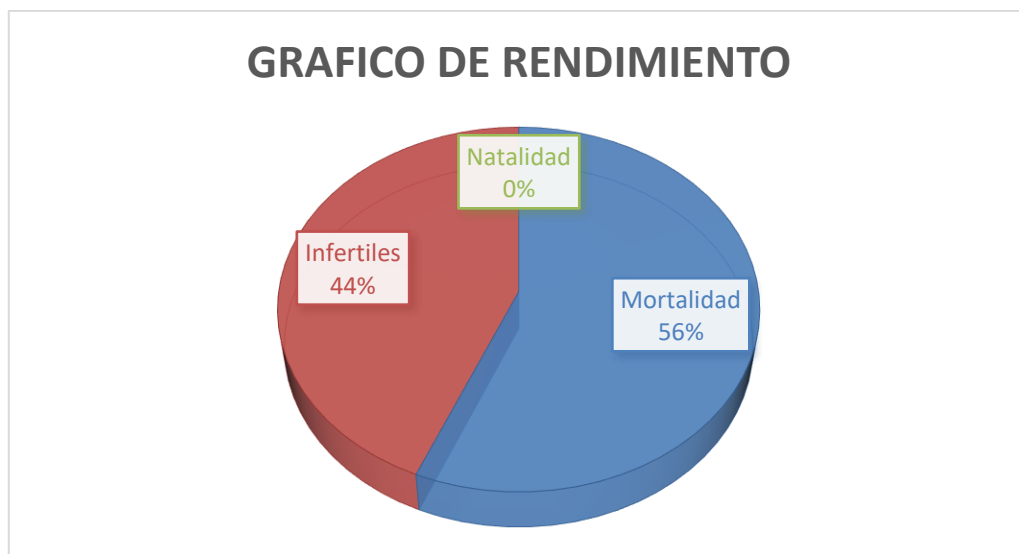


Figura 41. Grafica del rendimiento del sistema.

Adicionalmente, se identificó la necesidad de realizar una ovoscopia más precisa, lo cual conllevó cambios significativos en las variables ambientales. Estas intervenciones podrían haber influido en los resultados adversos obtenidos.

4.2 SEGUNDO PROCESO DE INCUBACIÓN

4.2.1 SELECCIÓN DE HUEVOS FÉRTILES

Para la selección de los huevos fértiles de igual manera que en el primer proceso de incubación se los obtuvo a través de la empresa Avícola Avigofri, los cuales se muestran en la siguiente Figura.



Figura 42. Colocación de los huevos fértiles en el prototipo Implementado.

4.2.2 INICIO DEL PROCESO DE INCUBACIÓN

Para iniciar el proceso de incubación en el prototipo se procedió a colocar cada uno de los huevos en la cubeta y esta a su vez en la cámara de incubación, igualmente se conectan los sensores y un ves cerrada la puerta se procede a iniciar su activación presionando el botón INICIAR en la pantalla.



Figura 43. Encendido y puesta en funcionamiento del prototipo.

4.2.3 PRUEBAS DE OVOSCOPIA

Una vez iniciado el proceso de incubación se realizó pruebas de ovoscopia para lo cual el prototipo cuenta con un sistema de iluminación independiente para cada huevo además de ser totalmente automático y prevenir la intervención innecesaria del operador. A continuación, se muestra un resumen de los datos obtenidos.

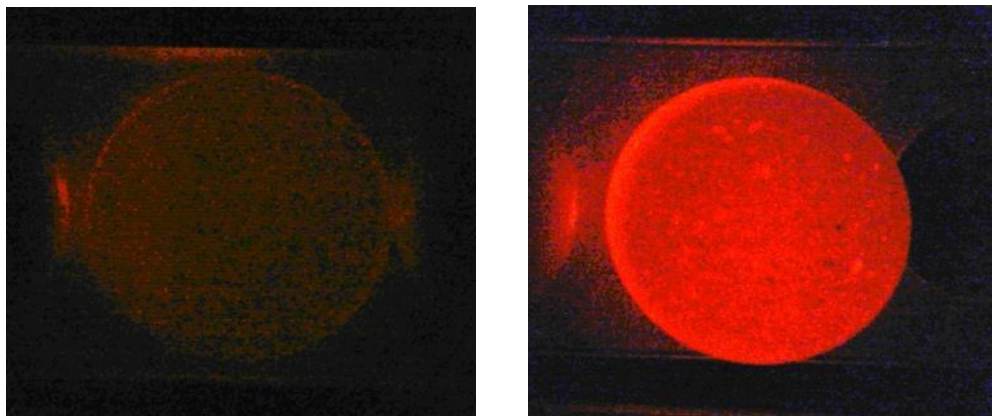


Figura 44. Primera prueba de ovoscopia en el día 7. A) Huevo Fértil. B) Huevo Infértil.

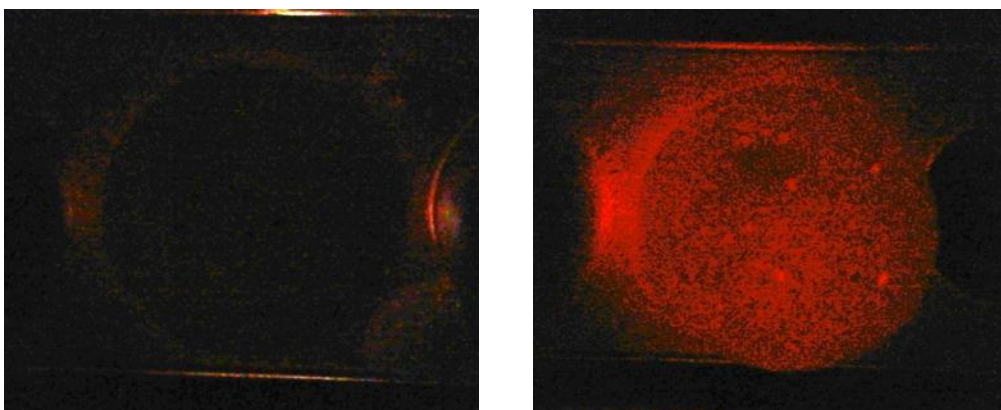


Figura 45. Segunda prueba de ovoscopia en el día 14. A) Huevo Fértil. B) Huevo Infértil.

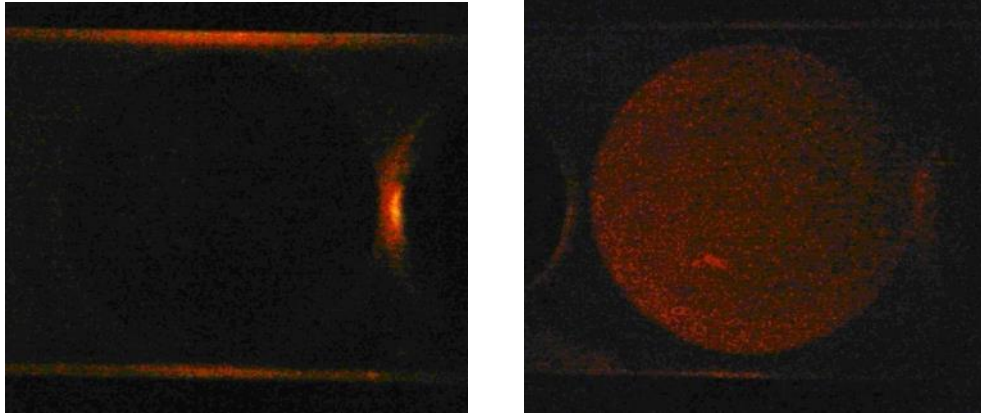


Figura 46. Última prueba de ovoscopia en el día 18. A) Huevo Fértil. B) Huevo Infértil.

4.2.4 FIN DE LA INCUBACIÓN

Una vez transcurridos los 21 días, se observó que los huevos comenzaron a eclosionar, a diferencia del anterior proceso en el cual no se observó ninguna señal de nacimiento de los polluelos.



Figura 47. Eclosión de huevos.



Figura 48. Polluelos obtenidos al final de la incubación.

4.2.5 ANÁLISIS DE LOS PARÁMETROS DE HUMEDAD Y TEMPERATURA

Se procede con el análisis de los valores obtenidos durante la incubación, a continuación, se muestra un resumen de los resultados obtenidos.

Tabla 16. Resumen de los valores de Temperatura y Humedad obtenidos en la incubación.

Dia	Temperatura	Humedad	Dato-SetPoint Temperatura	Dato-SetPoint Humedad
0	37,72	54,71	0,0	-5,3
1	37,62	57,19	-0,1	-2,8
2	37,47	58,48	-0,2	-1,5
3	37,48	57,79	-0,2	-2,2
4	37,50	58,06	-0,2	-1,9
5	37,57	57,94	-0,1	-2,1
6	37,49	57,47	-0,2	-2,5
7	37,50	58,45	-0,2	-1,6
8	37,50	58,38	-0,2	-1,6
9	37,53	58,04	-0,2	-2,0
10	37,47	57,95	-0,2	-2,0
11	37,52	58,20	-0,2	-1,8
12	37,47	57,76	-0,2	-2,2
13	37,57	57,16	-0,1	-2,8
14	37,51	57,85	-0,2	-2,1
15	37,47	56,55	-0,2	-3,4
16	37,45	58,00	-0,2	-2,0
17	37,48	57,91	-0,2	-2,1
18	37,45	57,40	-0,2	-2,6
19	36,65	62,09	0,2	-2,9
20	36,69	63,67	0,2	-1,3
21	36,65	63,75	0,1	-1,3

De manera similar al proceso anterior, se registraron y analizaron los datos generados durante la incubación del segundo sistema. Los resultados se presentan en la Tabla 16, y se procederá a realizar un análisis comparativo utilizando el mismo enfoque empleado previamente.

Tabla 17. Resultados estadísticos de la humedad.

Temperatura		
Dia 0 al 18		
Media	Desviación Estándar	Varianza
0.2 °C	0.0596°C	0..003
Dia 19 al 21		
Media	Desviación Estándar	Varianza

0.2°C	0.0247°C	0.00061
-------	----------	---------

En la Tabla 17, se observa que durante los primeros 18 días, se registraron valores cercanos a cero. Esto indica que el sistema de control logró mantener la temperatura en niveles adecuados, con cierta variabilidad en los datos. En los últimos 3 días, se observó una mayor estabilidad en la temperatura, con valores más próximos al nuevo setpoint establecido. Estos resultados indican una mejora en el control de la temperatura durante esta etapa del proceso de incubación.

Tabla 18. Resultados estadísticos humedad.

Humedad		
Dia 0 al 18		
Media	Desviación Estándar	Varianza
2.4%	0.86%	0.739
Dia 19 al 21		
Media	Desviación Estándar	Varianza
1.8%	0.935%	0.875

La Tabla 18 presenta los resultados estadísticos del análisis de la humedad. Durante los primeros 18 días, se observaron variaciones en los niveles de humedad en relación al setpoint establecido. Estos datos muestran cierta variabilidad en la humedad durante esta etapa. En los últimos 3 días, se registró una mayor dispersión en los valores de humedad en comparación con la etapa anterior. Estos resultados sugieren que el sistema de control experimentó dificultades para mantener la humedad en niveles estables durante esta fase final del proceso de incubación.

A continuación, se muestra las gráficas de comportamiento tanto de la temperatura como de la humedad.

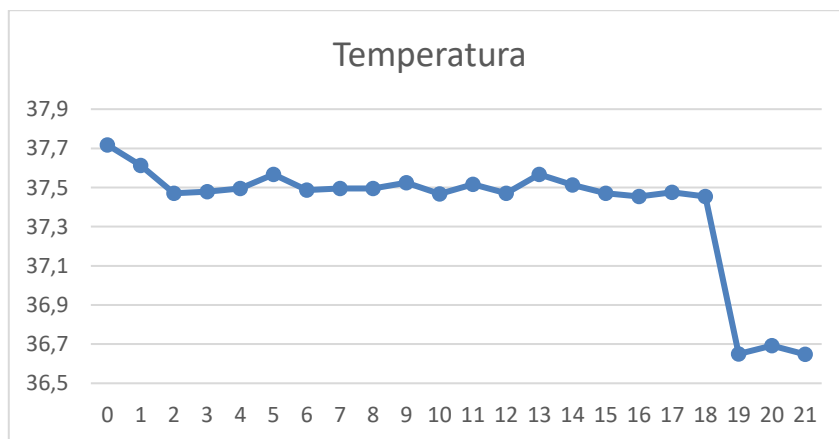


Figura 49. Grafica del comportamiento de la Temperatura.

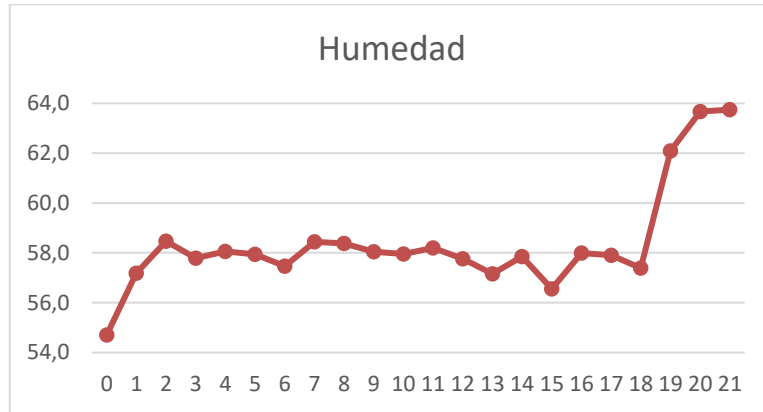


Figura 50. Grafica del comportamiento de la Humedad.

Con respecto al control de temperatura y humedad en el prototipo se pudo apreciar que, los valores de temperatura marcaban un error promedio de 0.18 °C teniendo una precisión más aceptable, de igual manera, la humedad se mantuvo con un error promedio de 2.1%. y aunque su tendencia era mantenerse por debajo del valor establecido, su variación no tubo cambios significativos.

4.2.6 ANÁLISIS DE RESULTADOS

De igual manera que en el proceso anterior procedemos al análisis de los resultados que se obtuvieron al final de la incubación.

Tabla 19. Resultados del rendimiento en la producción del Prototipo.

Total, de huevos Incubados	Huevos Infértiles	Huevos Fértiles	Mortalidad de Huevos	Natalidad de Huevos
57	21	36	25	11

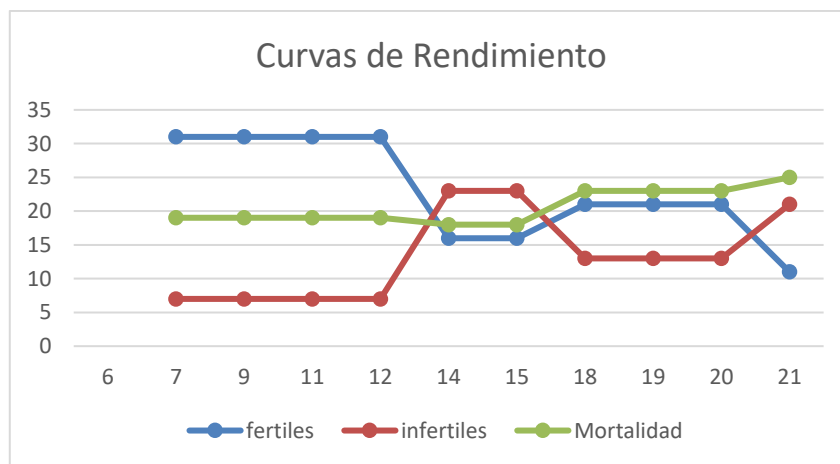


Figura 51. Grafica de la Curva de Rendimiento del Prototipo.

En la Figura 51 y Figura 52 se muestra los resultados obtenidos en este sistema. Se observa que, de los 36 huevos fértiles, únicamente 11 lograron eclosionar exitosamente. Esto

representa una eficiencia de incubación del 30.6%. Aunque se obtuvo una tasa de eclosión positiva en comparación con el sistema anterior, todavía existe margen de mejora para optimizar el proceso de incubación y aumentar la eficiencia reproductiva.

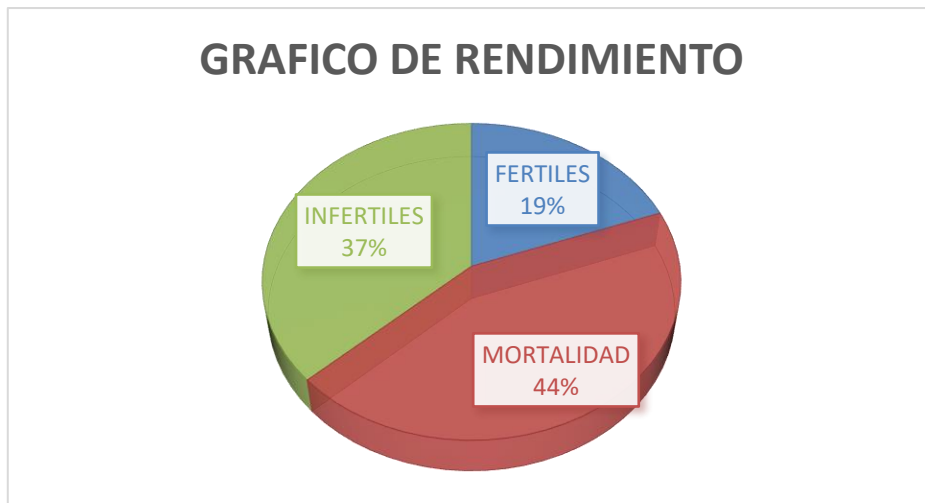


Figura 52. Gráfico de Rendimiento del Prototipo.

Es importante destacar que el sistema implementado ha demostrado un progreso en términos de reducción de huevos infértiles y un aumento en la tasa de eclosión en comparación con el sistema anterior.

4.3 ANÁLISIS GENERAL

Para realizar un análisis general de los sistemas se aplicó el test de Kolmogorov-Smirnov para evaluar la normalidad de los datos de temperatura y humedad en ambos sistemas de incubación. Las hipótesis planteadas fueron las siguientes:

- Hipótesis nula (H0): Los datos siguen una distribución normal.
- Hipótesis alternativa (H1): Los datos no siguen una distribución normal.

Para el primer sistema de incubación, los resultados del test de Kolmogorov-Smirnov para los datos de temperatura arrojaron:

Tabla 20. Resultados del test de normalidad para el sistema existente.

Temperatura		Humedad	
Parámetro	Valor	Parámetro	Valor
Media (x)	37,61727273	Media (x)	52,24363636
Desviación estándar	0,309918297	Desviación estándar	3,240312368
KS(c)	0,392865842	KS(c)	0,199425836
α	0,05	α	0,05
$c\alpha$	0,895	$c\alpha$	0,895

k(n)	4,861636369	k(n)	4,861636369
KS(t)	0,184094394	KS(t)	0,184094394
p-valor	0,00139	p-valor	0,30744

Al comparar los valores obtenidos de **KS(c)** con los valores críticos **KS(t)** para un nivel de significancia de 0.05, se observa que en ambos casos temperatura y humedad, los valores de **KS(c)** son mayores que los valores críticos correspondientes. Además, que su valor **p-valor** en el caso de la temperatura es mucho menor que el nivel de significancia α , y el **p-valor** de la humedad es mayor al nivel de significancia α , por lo tanto, se esto indica que se rechaza la hipótesis nula para la temperatura, lo que implica que los datos de temperatura no siguen una distribución normal. Sin embargo, en el caso de la humedad, no se dispone de suficiente evidencia para afirmar que los datos difieren significativamente de una distribución normal.

Tabla 21. Resultados del test de normalidad para el sistema implementado.

Temperatura		Humedad	
Parámetro	Valor	Parámetro	Valor
Media (x)	37,39674736	Media (x)	58,3990695
Desviación estándar	0,304543246	Desviación estándar	2,115925579
KS(c)	0,438410607	KS(c)	0,349323296
α	0,05	α	0,05
$c\alpha$	0,895	$c\alpha$	0,895
k(n)	4,861636369	k(n)	4,861636369
KS(t)	0,184094394	KS(t)	0,184094394
p-valor	0,00029	p-valor	0,000689

De manera similar, para el sistema implementado se obtuvieron valores de **KS(c)** mayores a los de **KS(t)** tanto en la temperatura y humedad y también se aprecia que el **p-valor** en los dos es menor al nivel de significancia α . Por lo cual se concluye que los datos no siguen una distribución normal.

Por lo tanto, con base en los resultados obtenidos y considerando el criterio de rechazo de la hipótesis nula, se puede afirmar parcialmente que los datos de temperatura y humedad en ambos sistemas de incubación no siguen una distribución normal.

Debido a que los datos obtenidos tanto en el primer sistema como en el sistema implementado se consideran que no siguen una distribución normal, se optó por utilizar una prueba no paramétrica, en este caso, el test de Wilcoxon, para realizar un análisis más adecuado de los datos recolectados, siendo planteadas las siguientes hipótesis:

- Hipótesis nula (**H0**): No hay diferencia significativa entre los valores recolectados de temperatura y humedad entre el sistema existente y el sistema desarrollado.
- Hipótesis alternativa (**H1**): Existe una diferencia significativa entre los valores recolectados de temperatura y humedad entre el sistema existente y el sistema desarrollado.

Obteniendo los siguientes resultados:

Tabla 22. Resultados del test de Wilcoxon

Temperatura						
Suma de ranking (+)	Suma de ranking (-)	Población N	Valor W	Valor crítico	Z(cal)	p-valor
250	3	22	3	66	-2.835	0.00229
Humedad						
Suma de ranking (+)	Suma de ranking (-)	Población N	Valor W	Valor crítico	Z(cal)	p-valor
2	251	22	2	66	-2.858	0.00213

En los dos casos tanto en la temperatura como en la humedad el valor obtenido de la suma de ranking (W) es menor al valor crítico, además sus valores de **p-valor** son menores al nivel de significancia de 0.05. Por lo tanto, se rechaza la hipótesis nula y se acepta la hipótesis alternativa.

En resumen, los resultados del test de Wilcoxon indican que existe una diferencia significativa entre los valores de temperatura y humedad recolectados entre el sistema existente y el sistema desarrollado. Esto respalda la afirmación de que hay una mejora o variación significativa en los parámetros de temperatura y humedad en el sistema desarrollado en comparación con el sistema existente.

CAPÍTULO V.

5 CONCLUSIONES y RECOMENDACIONES

5.1 CONCLUSIONES

- Se logró diseñar un sistema de control, el cual permitió controlar de manera eficiente la temperatura y humedad dentro de la cámara de incubación y un sistema de monitoreo automatizado el cual permitió la obtención de imágenes gracias al mecanismo de posicionamiento de la cámara sobre cada uno de los huevos, basado en dos motores y dos tornillos sin fin, empleando totalmente hardware y software libre.
- Se eligió algoritmos de detección de bordes y de selección de características los cuales permitieron generar un mejor monitoreo sobre la fertilidad de los huevos, sin embargo, la eficiencia de estos se encuentra limitada por dos factores principales como lo son: la iluminación con la cual entre mayor sea se obtendrá un mejor detalle de las características de los huevos y la cámara puesto que en base a su calidad es posible obtener una mayor nitidez de las imágenes. Por otro lado, para la comunicación se utilizó el protocolo MQTT y el servicio de base de datos de FIRESTORE de Google.
- Se realizó la implementación de un prototipo funcional de incubadora cumpliendo con los parámetros propuestos además de la implementación de una aplicación móvil desarrollada con Android Studio la cual permite el control y visualización de las variables ambientales y el monitoreo de los huevos en el prototipo.
- Los resultados de los tests realizados a cada sistema de incubación revelaron diferencias significativas en los valores de temperatura y humedad. Se observó que el sistema desarrollado logró mantener las condiciones ambientales dentro de los rangos requeridos de manera más eficiente, lo cual se reflejó en una mejora significativa en la natalidad de los polluelos. Estos hallazgos respaldan la eficacia y mejora del sistema implementado, demostrando su capacidad para optimizar la producción en los huevos.

5.2 RECOMENDACIONES

- Un factor primordial antes de iniciar con la incubación es estar seguro de que los huevos a incubar hayan tenido una selección y manejo adecuado, ya que dependiendo de ello se obtendrán resultados favorables en su natalidad.
- Para mejorar el rendimiento de la incubación se puede verificar el sistema de ventilación ya que de este dependen las variables de temperatura y humedad dentro de la cámara de incubación además de ser el responsable de renovar el aire y podría ser un causante directo de la mortalidad de los embriones.
- Si se desea obtener una mayor resolución en las imágenes para poder identificar con mayor detalle las características físicas de cada huevo, se puede considerar el uso de una cámara de mayor resolución, ya sea tipo USB o una específica para la Raspberry Pi. Sin embargo, es importante tener en cuenta que esto implicaría un costo adicional.
- Para garantizar el almacenamiento adecuado de una mayor cantidad de datos generados por los sensores durante el proceso de incubación, es importante considerar comprar un plan con mayor capacidad que el gratuito ofrecido por Firebase para evitar posibles limitaciones y pérdida de datos.
- Dado que la incubadora debe de funcionar de manera continua durante los 21 días que dura su proceso, se recomienda utilizar un sistema de alimentación ininterrumpida que permita mantener el dispositivo en funcionamiento durante un tiempo limitado en caso de un corte de energía eléctrica. Esto asegurará la continuidad del proceso de incubación y evitara posibles interrupciones o pérdidas de datos.

BIBLIOGRAFÍA

- [1] Heri *et al.*, “Design of monitoring and automation systems for greenhouse environment based on IoT,” *IOP Conf Ser Mater Sci Eng*, vol. 801, no. 1, 2020, doi: 10.1088/1757-899X/801/1/012096.
- [2] M. H. Lashari, A. A. Memon, S. A. A. Shah, K. Nenwani, and F. Shafqat, “IoT Based poultry environment monitoring system,” *Proceedings - 2018 IEEE International Conference on Internet of Things and Intelligence System, IOTAIS 2018*, pp. 1–5, 2019, doi: 10.1109/IOTAIS.2018.8600837.
- [3] C. Prabowo *et al.*, “The Implementation of IoT (Internet of Things) for Controlling Cow Health,” *IOP Conf Ser Mater Sci Eng*, vol. 846, no. 1, 2020, doi: 10.1088/1757-899X/846/1/012011.
- [4] R. Rakhmawati, Irianto, F. D. Murdianto, A. Luthfi, and A. Y. Rahman, “Thermal optimization on incubator using fuzzy inference system based IoT,” *Proceeding - 2019 International Conference of Artificial Intelligence and Information Technology, ICAIT 2019*, pp. 464–468, 2019, doi: 10.1109/ICAIT.2019.8834530.
- [5] L. K. S. Tolentino, E. G. Justine Enrico, R. L. M. Listanco, M. M. Anthony Ramirez, T. L. U. Renon, and M. B. Rikko Samson, “Development of Fertile Egg Detection and Incubation System Using Image Processing and Automatic Candling,” *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, vol. 2018-October, no. October, pp. 701–706, 2019, doi: 10.1109/TENCON.2018.8650320.
- [6] A. Zambrano, “Ganadería y avicultura: Importantes rubros en la economía,” *El Agro*, p. 68, Nov. 10, 2014. [Online]. Available: <https://issuu.com/jenn2808/docs/elagro-edicion219final-sep12.compre>
- [7] INEC, “Encuesta de Superficie y Producción Agropecuaria Continua 2020 Contenido,” Ecuador, May 2021. Accessed: Feb. 07, 2022. [Online]. Available: https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_ag
- [8] C. A. Barboza Pitre, “Diseño de una incubadora de huevos de gallina para las granjas productoras y comercializadoras avícolas en el municipio de Becerril-Cesar.,” 2021. Accessed: Mar. 14, 2022. [Online]. Available: <http://repository.uamerica.edu.co/handle/20.500.11839/8615>
- [9] E. Corti and E. Vogelaar, “THE OLDEST HATCHERIES ARE STILL IN USE.” Accessed: Sep. 05, 2022. [Online]. Available: <http://www.aviculture-europe.nl/nummers/12e03a08.pdf>
- [10] Á. A. Ramón Maldonado, “Diseño y construcción de un prototipo de incubadora con supervisión inteligente para la eclosión de huevos.,” *DSpace ESPOCH.*, Jul. 2019, Accessed: Feb. 06, 2022. [Online]. Available: <http://dspace.esepoch.edu.ec/handle/123456789/13693>

- [11] L. K. Tolentino, R. A. C. Alpay, A. J. N. Grutas, S. J. B. Salamanes, R. J. C. Sapiandante, and M. B. Vares, “An Automated Egg Incubator with Raspberry Pi-Based Camera Assisted Candling and R-CNN-based Maturity Detection,” *International Journal of Computing and Digital Systems*, vol. 11, no. 1, pp. 303–313, 2022, doi: 10.12785/ijcnds/110125.
- [12] “El porqué usar incubación de etapa única - Watt Industria Avícola.” <https://www.industriaavicola.net/reproduccion-genetica-e-incubacion/el-porque-usar-incubacion-de-etapa-unica/> (accessed Sep. 09, 2022).
- [13] “CONCEPTOS BÁSICOS DE INCUBACIÓN: Carga única vs Carga múltiple - Avicultura.” <https://avicultura.com/conceptos-basicos-de-incubacion-carga-unica-vs-carga-multiple/> (accessed Sep. 09, 2022).
- [14] “Incubadora Covatutto Mod.16.” <https://www.masalles.com/es/maquinaria-av%C3%ADcola/incubadoras/incubaci%C3%B3n-completa/96-incubadora-covatutto-mod16> (accessed Sep. 19, 2022).
- [15] criadeaves, “Tipos de Incubadoras para Huevos de Gallina,” *criadeaves*, Jan. 07, 2020. <https://criadeaves.com/gallinas-ponedoras/tipos-de-incubadoras-para-huevos/> (accessed Feb. 07, 2022).
- [16] C. E. Larrosa, “Cría de aves: cómo usar la incubadora familiar,” *EEA Santa Cruz, INTA*, Jun. 2017, Accessed: Aug. 27, 2022. [Online]. Available: <http://repositorio.inta.gob.ar:80/handle/20.500.12123/11281>
- [17] J. A. Cuéllar, “Incubación: obtención de pollitas para puesta y de pollitos para carne,” *veterinariadigital*, Jan. 07, 2021. https://www.veterinariadigital.com/articulos/incubacion-en-gallinas-ponedoras/#Factores_clave_de_la_incubacion (accessed Feb. 07, 2022).
- [18] J. C. Ganazhapa Malla, “Diseño y construcción de un prototipo de incubación artificial de huevos, con control automático de temperatura y humedad para la Avícola Ganazhapa, en la Parroquia Taquil de la Ciudad de Loja.,” Feb. 2017, Accessed: Sep. 09, 2022. [Online]. Available: <http://dspace.esPOCH.edu.ec/handle/123456789/6767>
- [19] I. C. Boleli, V. S. Morita, J. B. Matos, M. Thimotheo, and V. R. Almeida, “Poultry egg incubation: Integrating and optimizing production efficiency,” *Rev Bras Cienc Avic*, vol. 18, no. Special Issue 2, pp. 1–16, 2016, doi: 10.1590/1806-9061-2016-0292.
- [20] Emtech Hatchery Systems, “Volteo del huevo durante la Incubación – una revisión de Mohamed Mostafa El-Ashram - EmTech Hatchery Systems Ltd,” *Emtech Systems*, May 11, 2020. <https://www.emtech-systems.com/es/charlas-tecnicas/egg-turning-during-incubation-a-review-by-mohamed-mostafa-el-ashram/> (accessed Aug. 24, 2022).

- [21] Gallipedia, “Incubadora artificial,” *gallipedia*, 2017. <http://www.gallipedia.es/incubadora-artificial/> (accessed Feb. 07, 2022).
- [22] “Estructuras portantes de aluminio - Edificaciones Dinámicas.” <https://edificacionesdinamicas.com/estructuras-portantes-de-aluminio/> (accessed Sep. 20, 2022).
- [23] “Incubadora Modelo 2600-I HLC.” <https://www.masalles.com/es/maquinaria-av%C3%ADcola/incubadoras/incubaci%C3%B3n-completa/34-incubadora-modelo-2600-i-hlc> (accessed Sep. 21, 2022).
- [24] Sabelotodo, “Incubadora para huevos casera.” <http://www.sabelotodo.org/hagalousted/incubadora.html> (accessed Sep. 09, 2022).
- [25] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, “A vision of IoT: Applications, challenges, and opportunities with China Perspective,” *IEEE Internet Things J*, vol. 1, no. 4, pp. 349–359, Aug. 2014, doi: 10.1109/JIOT.2014.2337336.
- [26] V. Alvear-Puertas, P. Rosero-Montalvo, D. Peluffo-Ordóñez, and J. Pijal-Rojas, “Internet de las Cosas y Visión Artificial, Funcionamiento y Aplicaciones: Revisión de Literatura,” *Enfoque UTE*, vol. 8, no. 1, pp. 244–256, Feb. 2017, doi: 10.29019/ENFOQUEUTE.V8N1.121.
- [27] ENGINEERS GARAGE, “IoT Standards and Protocols: IoT Part 3.” <https://www.engineersgarage.com/iot-standards-and-protocols-iot-part-3/> (accessed Sep. 20, 2022).
- [28] J. D. C. Silva, J. J. P. C. Rodrigues, J. Al-Muhtadi, R. A. L. Rabêlo, and V. Furtado, “Management platforms and protocols for internet of things: A survey,” *Sensors (Switzerland)*, vol. 19, no. 3, Feb. 2019, doi: 10.3390/s19030676.
- [29] ENGINEERS GARAGE, “Application Layer Protocols for IoT: IoT Part 11.” <https://www.engineersgarage.com/application-layer-protocols-for-iot-iot-part-11/> (accessed Sep. 20, 2022).

ANEXOS

Anexo 1. Datos de Temperatura y humedad por día de la incubadora HDD mini 56S

DATOS DIA 0				
Hora	TEMPERATURA AHT10	HUMEDAD AHT10	TEMPERATURA INCUBADORA	HUMEDAD INCUBADORA
15:00	37,7	70,1	37,7	55,4
16:00	37,8	70,2	37,7	55,4
17:00	37,8	75,9	37,7	55,2
18:00	37,9	75,8	37,7	55,3
19:00	37,9	77	37,8	60
20:00	37,6	73,3	37,7	55,2
21:00	37,8	54,2	37,7	40,1
22:00	37,8	59,5	37,8	45
23:00	37,9	63,7	37,8	45
0:00	37,8	63,3	37,7	45
1:00	38,2	62,7	37,7	45,3
2:00	37,9	62,9	37,7	45,2
3:00	38,1	62,1	37,7	40,1
4:00	38,3	60,8	37,8	42,5
5:00	37,9	61,5	37,7	41,3
6:00	37,9	60,2	37,7	40,4
7:00	37,7	60,1	37,7	40,1
8:00	37,6	61,3	37,7	40,3
9:00	37,9	68,3	37,7	50,2
10:00	37,9	68,5	37,8	50,2

DATOS DIA 1				
Hora	TEMPERATURA AHT10	HUMEDAD AHT10	TEMPERATURA INCUBADORA	HUMEDAD INCUBADORA
11:00	37,7	69,7	37,7	50,1
12:00	37,7	69,5	37,7	50
13:00	37,7	70,4	37,7	50,1
14:00	37,8	55	37,8	40
15:00	37,8	64,5	37,7	45,2
16:00	37,9	64,7	37,7	45,1
17:00	37,7	65	37,7	45
18:00	37,6	64,6	37,7	45
19:00	37,9	63,2	37,7	45
20:00	37,8	64,1	37,8	45,3
21:00	37,8	57,2	37,7	40,3
22:00	37,8	55,8	37,8	40,1
23:00	37,9	62,3	37,7	45,1
0:00	37,8	61,8	37,7	45
1:00	38,3	58	37,7	40,4
2:00	38,1	57,5	37,7	40,3
3:00	37,8	58,1	37,7	40,4
4:00	37,9	56,7	37,8	40,3
5:00	37,9	56,6	37,7	40,1
6:00	37,9	56,3	37,7	40,1
7:00	37,9	52	37,7	35,5
8:00	37,9	63,1	37,7	45,2
9:00	37,7	65,8	37,8	45,3
10:00	38	62,4	37,7	45,2

DATOS DIA 2				
Hora	TEMPERATURA AHT10	HUMEDAD AHT10	TEMPERATURA INCUBADORA	HUMEDAD INCUBADORA
11:00	37,9	93,4	37,7	45,1
12:00	37,9	62	37,7	45,1
13:00	37,9	61,3	37,8	40,3
14:00	38	61	37,8	40,2
15:00	37,9	61,3	37,7	40,1
16:00	37,9	60,3	37,8	40,1
17:00	37,9	59,1	37,7	40
18:00	37,9	59,9	37,7	40,1
19:00	37,9	58,2	37,7	40,4
20:00	37,7	58,5	37,7	40,4
21:00	37,7	60,1	37,6	40,2
22:00	37,8	60,2	37,9	60,8
23:00	37,9	60,8	37,7	45
0:00	37,4	60	37,8	45,1
1:00	37,4	59,7	37,7	40,9
2:00	37,3	59,8	37,8	40,3
3:00	37,7	57,3	37,7	40,2
4:00	37,6	56,5	37,7	40,5
5:00	37,8	50,6	37,8	39,9
6:00	37,9	52,6	37,6	39,1
7:00	37,8	60,3	37,7	45
8:00	37,9	58,5	37,8	40,1
9:00	37,8	57,8	37,7	40
10:00	37,7	57,6	37,7	40,1

DATOS DIA 3				
Hora	TEMPERATURA AHT10	HUMEDAD AHT10	TEMPERATURA INCUBADORA	HUMEDAD INCUBADORA
11:00	37,9	57,6	37,7	40,1
12:00	37,8	56,9	37,7	40,5
13:00	37,7	61,8	37,7	45,1
14:00	37,9	58,1	37,8	40,4
15:00	37,7	60,5	37,7	40
16:00	37,7	58,7	37,8	40,1
17:00	37,8	58,6	37,7	40,2
18:00	37,7	59,2	37,7	40
19:00	37,7	57,1	37,8	40,3
20:00	37,6	58,8	37,7	40,1
21:00	37,7	58,3	37,7	40
22:00	37,7	58,2	37,8	40,1
23:00	37,8	59,4	37,8	41,7
0:00	37,9	57,6	37,7	40,6
1:00	37,2	59,1	37,7	40,1
2:00	37,3	58,6	37,8	40,1
3:00	37,9	57,9	37,8	40,4
4:00	37,8	56,9	37,7	40,2
5:00	37,4	56,9	37,7	40,3
6:00	37,6	56,6	37,7	40,2
7:00	38,1	56,2	37,8	40,2
8:00	37,6	59,3	37,7	40
9:00	37,5	60,3	37,7	40,1
10:00	37,8	59,6	37,7	40,1

DATOS DIA 4				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	60,1	37,8	40
12:00	37,7	58,3	37,7	40,1
13:00	37,8	60	37,7	40,2
14:00	37,8	64,8	37,8	45,2
15:00	37,8	65	37,8	50,2
16:00	37,7	64	37,8	45,2
17:00	37,8	63,8	37,7	45,2
18:00	37,7	62	37,8	45,3
19:00	37,8	62	37,8	45,2
20:00	37,7	62,2	37,8	45,2
21:00	37,7	60,9	37,7	45,2
22:00	37,8	61,5	37,7	45,1
23:00	37,7	60,8	37,8	45,2
0:00	37,5	60,3	37,8	45,3
1:00	37,8	58,6	37,8	40,1
2:00	37,7	57,3	37,8	40,6
3:00	37,4	58,7	37,8	40,1
4:00	37,2	58,1	37,7	40,1
5:00	37,8	56,4	37,7	40,3
6:00	37,8	55,5	37,7	40,2
7:00	37,7	54,3	37,8	40
8:00	37,8	58,7	37,7	40,3
9:00	37,9	55,4	37,8	40,4
10:00	37,7	54,9	37,8	40,1

DATOS DIA 5				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,6	52,3	37,7	35,6
12:00	37,7	52,2	37,7	35,7
13:00	37,7	50,9	37,7	35,5
14:00	37,9	56,5	37,8	40,2
15:00	37,8	58,4	37,7	40,5
16:00	37,7	57,7	37,7	40,4
17:00	37,8	57,6	37,7	40,5
18:00	37,7	57,8	37,7	40,4
19:00	37,8	56,1	37,7	40,2
20:00	37,7	45,7	37,7	35
21:00	38	63,6	37,7	45
22:00	37,8	65,2	37,7	45
23:00	37,8	57,6	37,7	40,3
0:00	37,2	59,6	37,8	40
1:00	37,3	59,3	37,8	40,1
2:00	37,9	57,5	37,7	40,4
3:00	37,9	56,1	37,7	40,2
4:00	37,4	56,5	37,8	40,3
5:00	37,6	56,2	37,7	40,3
6:00	37,7	56,5	37,7	41
7:00	37,7	58,8	37,8	40
8:00	37,8	56,4	37,7	40,3
9:00	37,8	64,3	37,7	45
10:00	37,7	64,6	37,7	45

DATOS DIA 6				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	63,8	37,7	45
12:00	37,8	62,7	37,7	45,3
13:00	37,7	62,9	37,8	45,3
14:00	37,7	61	37,7	45,1
15:00	37,7	61,5	37,8	45,1
16:00	37,7	59,4	37,8	40,2
17:00	37,7	60,1	37,8	45
18:00	37,9	61,2	37,8	45,1
19:00	37,7	62	37,8	45,1
20:00	37,7	59,2	37,8	45
21:00	38	59,9	37,7	45,1
22:00	37,2	61,4	37,8	45,3
23:00	37,3	60,9	37,8	45,3
0:00	36,9	58,2	37,8	40,1
1:00	37	51,7	37,8	35,9
2:00	37,2	46,7	37,7	35,3
3:00	37	46,1	37,7	35,2
4:00	37,4	45,2	37,8	30,9
5:00	37,5	62,5	37,7	45,3
6:00	37,2	62,7	37,8	45
7:00	37,3	61,7	37,7	45,4
8:00	37,7	58,5	37,7	40,1
9:00	37,7	53,9	37,7	40,2
10:00	37,7	53	37,7	40,2

DATOS DIA 7				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	52,8	37,7	40
12:00	37,7	52,6	37,7	35,8
13:00	37,7	70	37,7	55,3
14:00	37,7	64,4	37,8	45
15:00	37,7	58,2	37,8	40,6
16:00	37,7	60,1	37,7	40,2
17:00	37,7	60	37,8	40,2
18:00	37,8	59,2	37,8	40,2
19:00	37,8	58,3	37,8	40,4
20:00	38,1	53,4	37,8	35,6
21:00	37,9	60,5	37,8	40
22:00	37,4	61,3	37,7	45,4
23:00	37,4	60,5	37,7	45,3
0:00	37,1	57,2	37,7	40,5
1:00	37	49,7	37,7	35,6
2:00	37,2	47,3	37,7	35,3
3:00	37,1	47,1	37,7	35,2
4:00	37,6	60,4	37,8	40,2
5:00	37,4	61,6	37,7	45,3
6:00	37,2	62,7	37,8	45,6
7:00	37,6	57	37,8	40,2
8:00	37,3	60,6	37,6	40,5
9:00	37,7	55,3	37,7	40
10:00	37,8	55,3	37,8	40,1

DATOS DIA 8				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,7	55,9	37,7	40,1
12:00	37,8	54,5	37,8	40
13:00	37	56,01	37,7	40
14:00	37,7	68,2	37,7	50,1
15:00	37,7	63,8	37,8	45,3
16:00	37,7	60,4	37,7	45
17:00	37,7	63,4	37,8	45,3
18:00	37,7	60,1	37,7	45
19:00	37,8	60,8	37,7	45
20:00	38	60,2	37,9	40,3
21:00	37,6	63	37,7	45,1
22:00	37,7	65,1	37,7	45
23:00	38,1	62,2	37,7	45,2
0:00	37,9	62,7	37,7	45,1
1:00	37,3	64,5	37,8	45,3
2:00	37,2	61,6	37,8	40,2
3:00	38,1	49,5	37,7	35,5
4:00	37,7	61,7	37,7	40,2
5:00	37,7	66,3	37,7	45
6:00	38	61,8	37,7	45,2
7:00	37,9	59,6	37,7	40,1
8:00	37,9	58,7	37,8	40
9:00	37,7	59	37,8	40,5
10:00	37,7	58,9	37,7	40,4

DATOS DIA 9				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,7	58,7	37,7	40,5
12:00	37,7	57,8	37,7	40,3
13:00	37	56	37,7	40
14:00	37,7	63,7	37,7	50,2
15:00	37,8	62,5	37,7	50,1
16:00	37,6	61,9	37,7	45,3
17:00	37,8	58,9	37,8	45,3
18:00	37,9	60,3	37,7	40,4
19:00	37,7	66,3	37,7	45,3
20:00	37,9	66,4	37,7	49,8
21:00	37,7	69,1	37,6	50
22:00	37,7	55	37,7	40
23:00	37,9	60,8	37,7	45,1
0:00	37,4	60	37,6	45
1:00	37,9	60,5	37,6	49,8
2:00	37,5	61,5	37,8	40,3
3:00	37,3	60,4	37,7	40
4:00	37,7	59,8	37,6	45,3
5:00	37,3	59,7	37,6	40
6:00	37,9	58,9	37,7	40,1
7:00	37,9	58,2	37,7	40
8:00	37,6	61,3	37,7	45
9:00	37,7	61,6	37,7	45,1
10:00	37,6	60,4	37,7	40,2

DATOS DIA 10				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,9	58,6	37,7	40
12:00	37,8	54,5	37,8	40
13:00	37,7	59,7	37,8	40
14:00	37,7	58,6	37,7	40,4
15:00	37,7	63,8	37,8	45,3
16:00	37,7	76,3	37,8	55,1
17:00	37,8	74,8	37,7	55,3
18:00	37,7	76,5	37,7	60
19:00	37,8	75,8	37,7	60,1
20:00	37,9	54,5	37,7	40,5
21:00	37,8	62,8	37,7	45
22:00	37,2	70,1	37,3	50
23:00	37,9	65,2	37,7	50,2
0:00	37,3	61,3	37,7	45,1
1:00	37,5	61,1	37,7	45,2
2:00	37,3	61,2	37,8	40,3
3:00	37,5	61	37,7	35,5
4:00	37,4	60,5	37,8	40,1
5:00	37,5	60,7	37,7	45
6:00	37,7	62,2	37,8	45
7:00	37,7	61,5	37,8	45,3
8:00	37,8	68,2	37,8	50,3
9:00	37,7	66,1	37,7	50
10:00	37,8	68,8	37,8	51,3

DATOS DIA 11				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	68,2	37,7	50
12:00	37,8	63,7	37,7	50
13:00	37,7	63,6	37,8	50,2
14:00	37,9	62,6	37,7	50,2
15:00	37,9	63,4	37,8	50,3
16:00	37,7	63,1	37,7	50,1
17:00	37,9	58,7	37,7	45,3
18:00	37,8	58,8	37,8	45,3
19:00	37,7	57,3	37,7	40,3
20:00	37,7	58,1	37,8	45
21:00	37,8	55,3	37,8	40,2
22:00	37,6	55,3	37,8	40
23:00	37,7	59,9	37,7	40,1
0:00	37,7	58,3	37,7	45,1
1:00	37,9	57,5	37,7	45
2:00	37,8	58,1	37,7	45,2
3:00	37,8	57,4	37,6	40,1
4:00	37,9	57,1	37,7	40,5
5:00	38	56,4	37,8	45,4
6:00	37,9	56,6	37,8	45,4
7:00	37,7	58,7	37,7	45
8:00	37,9	56	37,7	40
9:00	37,9	62	37,7	45,1
10:00	37,7	62,2	37,8	45,1

DATOS DIA 12				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,9	57,3	37,8	40,1
12:00	37,8	57,6	37,8	40,1
13:00	37,7	59,6	37,8	45,1
14:00	37,8	63,8	37,8	45
15:00	37,6	65,4	37,7	45,2
16:00	37,7	62,5	37,7	45,2
17:00	37,8	58,5	37,7	40,1
18:00	37,4	70,7	37,6	50,4
19:00	37,6	57,3	37,7	40,1
20:00	37,7	63,6	37,8	40,2
21:00	37,9	59,1	37,8	40
22:00	37,8	59,1	37,7	40
23:00	37,6	55,6	37,7	40
0:00	37,7	53,1	37,8	35,9
1:00	37,7	55,2	37,7	40
2:00	37,7	55,4	37,8	40
3:00	37,9	52,1	37,7	35,8
4:00	37,7	56	37,8	40,1
5:00	37,6	55,3	37,7	45,2
6:00	37,7	54,2	37,8	45
7:00	37,5	52,9	37,8	45,3
8:00	37,7	57,4	37,7	40,2
9:00	37,8	61,1	37,8	45,2
10:00	37,7	60,5	37,7	45,3

DATOS DIA 13				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	60,3	37,7	45,1
12:00	37,6	59,1	37,8	40,2
13:00	37,7	60,1	37,8	45
14:00	37,6	59,4	37,8	45,1
15:00	37,9	53,7	37,7	40,2
16:00	37,8	59,3	37,8	45
17:00	37,9	57,6	37,7	40,1
18:00	38	57,3	37,8	40,1
19:00	37,6	55,6	37,8	35,8
20:00	37,6	63,1	37,7	40,2
21:00	37,8	58,3	37,7	40
22:00	37,7	52,3	37,8	39,8
23:00	37,5	64,5	37,7	45,1
0:00	37,8	64,5	37,7	45
1:00	37,9	64,2	37,8	45,4
2:00	37,8	63,5	37,8	45,4
3:00	37,9	63,5	37,7	45,2
4:00	37,8	62,9	37,7	45
5:00	37,5	65,2	37,7	45
6:00	37,9	61,2	37,8	45
7:00	37,9	62,9	37,8	40,3
8:00	37,4	61,1	37,7	40,1
9:00	37,5	63,2	37,7	45,2
10:00	37,6	63,8	37,8	45,4

DATOS DIA 14				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	60	37,7	45
12:00	37,7	56,8	37,8	40,5
13:00	37,5	63,8	37,7	45
14:00	37,9	60,3	37,5	50
15:00	37,8	58,9	37,7	45,1
16:00	37,7	59,4	37,7	45
17:00	37,8	58	37,7	40,1
18:00	37,9	57,5	37,7	40,1
19:00	37,9	56,1	37,7	40,4
20:00	37,8	55,9	37,8	40,4
21:00	37,6	52,6	37,6	40
22:00	37,8	55,5	37,8	40,3
23:00	37,6	60,8	37,8	40,5
0:00	37,6	55,2	37,7	40,2
1:00	37,6	54,5	37,7	45,1
2:00	37,7	54,2	37,7	45,2
3:00	37,6	53,8	37,7	45,4
4:00	37,6	52,8	37,8	45
5:00	37,6	52,7	37,8	45
6:00	37,6	58,4	37,8	40,3
7:00	37,6	59,6	37,7	40,2
8:00	37,4	60,1	37,7	40,3
9:00	37,7	60,8	37,7	40,1
10:00	37,7	61,3	37,7	40,1

DATOS DIA 15				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	60,7	37,8	40
12:00	37,7	61,3	37,7	40,1
13:00	38	57,5	37,7	40,2
14:00	37,8	52,4	37,7	40,2
15:00	36,6	67,8	37,7	45
16:00	37,6	61,7	37,8	45,2
17:00	37,6	61,5	37,7	45,2
18:00	37,7	63,2	37,7	45,2
19:00	37,7	63,1	37,7	45,2
20:00	37,9	65,1	37,7	45
21:00	37,9	61,7	37,7	40,2
22:00	37,9	61,5	37,7	40,1
23:00	37,5	60,7	37,7	40
0:00	37,6	56,5	37,7	40
1:00	37,7	54,8	37,8	35,8
2:00	37,6	53,5	37,7	35,5
3:00	37,6	52,2	37,7	35,4
4:00	37,5	56,6	37,6	35,8
5:00	37,6	54,7	37,7	35,5
6:00	37,4	50,9	37,8	35
7:00	37,7	60,1	37,7	40
8:00	38,1	64,7	37,7	45,3
9:00	37,8	64,8	37,7	45,6
10:00	37,9	65,2	37,8	45

DATOS DIA 16				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,7	65,5	37,7	45
12:00	37,8	64,3	37,7	45
13:00	37,9	62,2	37,8	45,4
14:00	37,9	62,2	37,8	45,3
15:00	37,9	62,2	37,7	45,2
16:00	37,9	61,9	37,7	45,2
17:00	37,8	61,4	37,8	45,2
18:00	37,7	63,2	37,7	40,3
19:00	37,7	63,4	37,6	40,3
20:00	37,6	63,7	37,7	40,2
21:00	37,8	58,5	37,7	40
22:00	37,9	61,2	37,7	40,3
23:00	37,6	60,8	37,7	40
0:00	37,5	64	37,7	45,1
1:00	37,6	63,7	37,7	45
2:00	37,6	63,3	37,7	45
3:00	37,6	62,1	37,8	45
4:00	37,6	62,9	37,7	45,4
5:00	37,6	62,6	37,7	45,3
6:00	37,6	62,3	37,8	45,4
7:00	37,7	63	37,8	40,1
8:00	37,8	62,3	37,7	45
9:00	37,9	62,9	37,7	45
10:00	37,8	61,2	37,7	40,1

DATOS DIA 17				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,8	58	37,7	40,3
12:00	37,7	58,7	37,7	40,3
13:00	37,7	58,8	37,7	40,4
14:00	37,9	58,2	37,8	40,3
15:00	37,7	52,2	37,7	35,5
16:00	37,9	51,5	37,7	35,4
17:00	37,6	55,8	37,6	35,9
18:00	37,8	52,4	37,7	35,4
19:00	37,7	53,2	37,6	39,8
20:00	37,9	55,8	37,7	40,1
21:00	37,8	59,9	37,7	40,5
22:00	37,7	60,5	37,7	45
23:00	37,7	61,1	37,7	45,3
0:00	37,6	60,8	37,8	45,2
1:00	37,6	60,5	37,7	45,2
2:00	37,8	60,5	37,8	45,2
3:00	37,7	60,2	37,8	45,3
4:00	37,7	60,4	37,7	45,2
5:00	37,6	60	37,7	45,3
6:00	37,6	59,7	37,7	45
7:00	37,7	56,4	37,7	40,1
8:00	37,8	61,7	37,7	45,1
9:00	37,8	62,5	37,8	45,3
10:00	37,6	61,1	37,8	40,2

DATOS DIA 18				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,9	58,5	37,8	40,2
12:00	37,7	57,9	37,7	40
13:00	37,5	58,3	37,6	40,3
14:00	37,6	58,1	37,8	40,3
15:00	37,7	56	37,8	40,4
16:00	37,7	56,8	37,7	40,1
17:00	37,6	56,8	37,8	40,1
18:00	37,9	55,3	37,7	40,2
19:00	37,9	54,4	37,8	40,2
20:00	37,8	55	37,7	40,3
21:00	37,6	54,1	37,7	39,6
22:00	37,9	55	37,7	40,2
23:00	37,6	52,6	37,7	35,9
0:00	37,5	51,9	37,7	35,5
1:00	37,4	63,1	37,4	45,1
2:00	37,5	61,9	37,5	44,2
3:00	37,6	61,3	37,7	45
4:00	37,5	61,2	37,7	45,1
5:00	37,5	60,9	37,8	45
6:00	37,4	59,4	37,5	40,3
7:00	37,5	58,3	37,5	40,3
8:00	37,4	56	37,6	40
9:00	37,6	64,2	37,5	45,1
10:00	37,6	64,4	37,5	45,2

DATOS DIA 19				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,6	64,1	37,5	45,1
12:00	37,6	63,5	37,5	45,2
13:00	37,6	64,6	37,6	45,3
14:00	37,6	66,2	37,5	45,1
15:00	37,9	64,2	37,5	40,1
16:00	37,7	63,2	37,7	40,3
17:00	37,6	63,2	37,6	45
18:00	37,9	65,5	37,7	45
19:00	37,6	66,3	37,7	45,1
20:00	37,6	65,5	37,5	45,1
21:00	37,4	65,8	37,5	40,5
22:00	37,5	65,2	37,5	45
23:00	37,4	60,4	37,4	45,2
0:00	37,7	64,1	37,7	50,3
1:00	37,5	61,5	37,7	50
2:00	37,6	61,3	37,8	45,3
3:00	37,5	60,4	37,8	50,2
4:00	37,7	66,4	37,7	45
5:00	37,3	64,1	37,7	45,3
6:00	37,2	65,1	37,5	45,2
7:00	37,5	65,3	37,5	45,2
8:00	37,5	65	37,5	45,4
9:00	37,7	64,7	37,5	45,4
10:00	37,6	64,3	37,6	45,2

DATOS DIA 20				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	37,7	63,3	37,5	45,1
12:00	37,6	63,2	37,6	45
13:00	37,6	61,7	37,6	40,1
14:00	37,9	58,2	37,6	40,3
15:00	37,7	59	37,5	40,5
16:00	37,4	59,1	37,6	40,5
17:00	37,4	57,4	37,5	40,3
18:00	37,4	56,8	37,6	40,2
19:00	37,2	68,4	37,5	45,4
20:00	36,4	70,7	36,6	50
21:00	36,5	70,5	36,6	50,3
22:00	36,6	70,2	36,6	50
23:00	36,3	71,2	36,7	50,1
0:00	36,3	70	36,6	50
1:00	36,2	68	36,7	45
2:00	36,3	69,6	36,7	45,1
3:00	36,3	68,6	36,7	45
4:00	36,2	69,2	36,6	45,5
5:00	36,2	68	36,6	50,1
6:00	36,2	69,2	36,6	50
7:00	36,3	69,6	36,7	50
8:00	36,4	69,5	36,6	45
9:00	36,3	68,6	36,6	45
10:00	36,3	68,8	36,6	50

DATOS DIA 21				
Hora	TEMPERATURA	HUMEDAD	TEMPERATURA	HUMEDAD
	AHT10	AHT10	INCUBADORA	INCUBADORA
11:00	36,4	70,7	36,6	50,2
12:00	36,2	73	36,7	50,1
13:00	36,3	72,3	36,7	50
14:00	36,5	71,4	36,6	50,2
15:00	36,5	72,3	36,6	50,2
16:00	36,5	71,3	36,6	50,2
17:00	36,5	71,4	36,6	50,2
18:00	36,5	70,2	36,6	55,1
19:00	36,1	87,1	36,6	50,3
20:00	36,2	82,5	36,6	50,1
21:00	36,5	83,5	36,6	50,3

Anexo 2. Datos de Temperatura y humedad por dia del prototipo.

DATOS DIA 0		
Hora	Temperatura	Humedad
04:35:51	37,6	60,7
05:35:51	37,9	53,6
06:35:51	37,8	59,3
07:35:51	37,5	52,5
08:35:51	37,9	54,9
09:35:51	37,7	54,6
10:35:51	37,4	56
11:35:51	37,8	53,9
12:35:51	37,8	54,2
13:35:51	37,8	54,3
14:35:51	37,8	54,5
15:35:51	37,8	54,1
16:35:51	37,8	53,8
17:35:51	37,8	53,1
18:35:51	37,6	52,5
19:35:51	37,6	52,6
20:35:51	37,6	56,4
21:35:51	37,8	56,8
22:35:51	37,8	54,9
23:35:51	37,6	54,7
00:35:51	37,7	54,2
01:35:51	37,7	53,8
02:35:51	37,7	52,9

DATOS DIA 1		
Hora	Temperatura	Humedad
03:35:51	37,9	52,1
04:35:51	37,7	51,1
05:35:51	37,4	59,6
06:35:51	37,7	56,7
07:35:51	37,8	57
08:35:51	36,7	62,4
09:35:51	37,6	65,1
10:35:51	38	60,9
11:35:51	37,2	60
12:35:51	37,9	57,6
13:35:51	37,8	57,7
14:35:51	37,6	57,8
15:35:51	38	57,3
16:35:51	37,8	58
17:35:51	37,9	57,4
18:35:51	37,8	62
19:35:51	37,5	56,1
20:35:51	37,5	58,5
21:35:51	37,5	57,4
22:35:51	37,3	55,2
23:35:51	37,5	54,3
00:35:51	37,6	53,2
01:35:51	37,5	52,8
02:35:51	37,5	52,3

DATOS DIA 2		
Hora	Temperatura	Humedad
03:35:51	37,5	61,9
04:35:51	37,5	61,4
05:35:51	37,5	61
06:35:51	37,1	57,1
07:35:51	37,5	58
08:35:51	37,7	57,8
09:35:51	37,3	55,7
10:35:51	37,7	56,9
11:35:51	37,7	56,4
12:35:51	37,6	56,8
13:35:51	37,9	58,4
14:35:51	37,4	59,4
15:35:51	38	59,1
16:35:51	37,4	58
17:35:51	37,8	57,1
18:35:52	37,9	58,2
19:35:51	37,4	57,4
20:35:52	37,5	56,5
21:35:51	37	58,4
22:35:52	37,2	57,7
23:35:51	37,1	60,1
00:35:51	37,2	60,5
01:35:51	37,2	60,2
02:35:52	37,2	59,4

DATOS DIA 3		
Hora	Temperatura	Humedad
03:35:51	37,2	58,8
04:35:51	37,2	58,7
05:35:51	36,9	56,7
06:35:52	37,4	56,7
07:35:51	37,3	55,5
08:35:52	37,6	56,1
09:35:52	37,4	57
10:35:51	37,5	55
11:35:51	37,5	57,9
12:35:51	37,5	57,4
14:35:51	37,3	58,6
15:35:51	37,2	55
16:35:51	37,2	59,1
17:35:51	37,7	58,9
18:35:51	37,9	58,9
19:35:51	37,3	59,5
20:35:51	37,8	60
21:35:51	37,8	59,6
22:35:51	37,9	58,9
23:35:51	37,7	57
00:35:51	37,4	59,3
01:35:51	37,9	57,9
02:35:51	37,4	56,6
13:35:51	37,5	57,8

DATOS DIA 4		
Hora	Temperatura	Humedad
03:35:51	37,7	59,4
04:35:51	37,4	59,9
05:35:51	37,5	60,1
06:35:51	37,2	52,1
07:35:51	37,7	59,7
08:35:51	37,6	60,4
09:35:51	37,2	59
10:35:51	37,3	56,1
11:35:51	37,5	59,3
12:35:51	37,4	59,2
13:35:51	37,7	59,7
14:35:51	37,4	57,4
15:35:51	37,5	57,7
16:35:51	37,4	59
17:35:51	37,8	58,2
18:35:51	37,7	57,7
19:35:51	37,7	57,7
20:35:51	37,2	55,4
21:35:51	37,4	58,1
22:35:51	37,5	57,4
23:35:51	37,6	57,9
00:35:51	37,4	57
01:35:51	37,8	56,6
02:35:51	37,3	58,5

DATOS DIA 5		
Hora	Temperatura	Humedad
03:35:51	37,7	56,5
04:35:51	37,4	58,4
05:35:51	37,5	58
06:35:51	37,2	56,9
07:35:51	37,8	58,2
08:35:51	37,1	59,2
09:35:51	37,4	56,8
10:35:51	37,8	60,6
11:35:51	37,4	56,7
12:35:51	37,6	59,6
13:35:51	37,5	56,8
14:35:51	37,8	56,9
15:35:51	37,7	58,8
16:35:51	37,2	57,3
17:35:51	37,4	57,9
18:35:51	37,8	58,8
19:35:51	37,9	57,1
20:35:51	38	56,4
21:35:51	37,6	59,6
22:35:51	37,6	56,8
23:35:51	37,5	59,6
00:35:51	37,8	59,1
01:35:51	37,5	57,3
02:35:51	37,4	57,3

DATOS DIA 6		
Hora	Temperatura	Humedad
03:35:51	37,5	58,2
04:35:51	37,6	59,6
05:35:51	37,1	57,5
06:35:51	37,5	58,3
07:35:51	37,4	56,2
08:35:51	37,6	59
09:35:51	37,8	57,3
10:35:51	37,4	60
11:35:51	37,8	59,1
12:35:51	37,7	55,6
13:35:51	37,4	57,5
14:35:51	37,5	58
15:35:51	37	57,1
16:35:51	37,2	56,6
17:35:51	37,5	57,4
18:35:51	37,5	52,3
19:35:51	37,8	57,8
20:35:51	37,1	58,5
21:35:51	37,5	57,3
22:35:51	37,4	57,9
23:35:51	37,3	59,5
00:35:51	37,8	57,1
01:35:51	37,7	56,2
02:35:51	37,6	55,2

DATOS DIA 7		
Hora	Temperatura	Humedad
03:35:51	37,5	58,2
04:35:51	37,2	58,9
05:35:51	37,6	59,8
06:35:51	37,7	59,4
07:35:51	37,9	59,2
08:35:51	37,4	57,5
09:35:51	37,4	58,8
10:35:51	37,5	59,4
11:35:51	37,1	58,3
12:35:51	37,3	59,6
13:35:51	37,4	56,9
14:35:51	37,2	59,4
15:35:51	37,1	60,7
16:35:51	37,9	59,4
17:35:51	37,8	57,5
18:35:51	37,3	60
19:35:51	37,5	56,3
20:35:51	37,5	55,2
21:35:51	37,4	58,2
22:35:51	38	57,8
23:35:51	37,4	58,8
00:35:51	37,4	56,3
01:35:51	38	58,7
02:35:51	37,4	58,4

DATOS DIA 8		
Hora	Temperatura	Humedad
03:35:51	37,6	57,6
04:35:51	37,3	56,6
05:35:51	37,4	58
06:35:51	37,7	58,1
07:35:51	37,5	57,9
08:35:51	37,8	58,7
09:35:51	37,4	60,6
10:35:51	37,6	60,2
11:35:51	37,4	58,9
12:35:51	37,5	59,4
13:35:51	37,2	59,7
14:35:51	37,5	56,3
15:35:51	37,8	60,2
16:35:51	37,2	58
17:35:51	37,6	57,9
18:35:51	37,8	57
19:35:51	37,4	58,3
20:35:51	37,5	58,5
21:35:51	37,6	58,5
22:35:51	37,4	58,9
23:35:51	37,6	57,3
00:35:51	37,4	57,3
01:35:51	37,4	59,5
02:35:51	37,3	57,6

DATOS DIA 9		
Hora	Temperatura	Humedad
03:35:51	37,4	57,1
04:35:51	37,5	59,2
05:35:51	37,4	57,8
06:35:51	37,5	58,3
07:35:51	37,6	54,6
08:35:51	37,7	56,2
09:35:51	37,6	57,6
10:35:51	38	56,2
11:35:51	37,5	57,2
12:35:51	37,4	59,1
13:35:51	37,4	59,1
14:35:51	37,4	56,2
15:35:51	37,8	57,9
16:35:51	37,5	59,4
17:35:51	37,5	59,5
18:35:51	37,5	59,5
19:35:51	37,5	57,9
20:35:51	37,6	60,2
21:35:51	37,4	58,3
22:35:51	37,4	59,1
23:35:51	37,9	56,6
00:35:51	37,3	59,8
01:35:51	37,4	59,5
02:35:51	37,4	56,7

DATOS DIA 10		
Hora	Temperatura	Humedad
03:35:51	37,4	60,3
04:35:51	37,3	56,4
05:35:51	37,5	60,1
06:35:51	37,5	59,1
07:35:51	37,5	55,4
08:35:51	37,9	56,3
09:35:51	37,5	58,9
10:35:51	37,1	57,1
11:35:51	37,7	57,3
12:35:51	37,6	59,8
13:35:51	37,1	59,1
14:35:51	37,5	60
15:35:51	37,5	58,1
16:35:51	37,6	58,8
17:35:51	37,5	55,7
18:35:51	37,3	57
19:35:51	37,6	56,9
20:35:51	37,5	56,4
21:35:51	37,5	54,9
22:35:51	37,4	57,3
23:35:51	37,5	59,4
00:35:51	37,3	57,4
01:35:51	37,5	59,9
02:35:51	37,4	59,3

DATOS DIA 11		
Hora	Temperatura	Humedad
03:35:51	37,4	59,9
04:35:51	37,6	59,7
05:35:51	37,4	58,5
06:35:51	37,5	59,6
07:35:51	37,7	56
08:35:51	37,5	57,6
09:35:51	37,3	57,2
10:35:51	37,8	58,9
11:35:51	37,8	57,3
12:35:51	37,4	58,2
13:35:51	37,7	57,8
14:35:51	37,2	56,5
15:35:51	37,5	59,2
16:35:51	37,7	57,1
17:35:51	37,3	57
18:35:51	37,5	57,8
19:35:51	37,8	57,2
20:35:51	37,3	58,8
21:35:51	37,5	58,4
22:35:51	37,7	59,1
23:35:51	37,5	57,2
00:35:51	37,4	57,9
01:35:51	37,5	59,6
02:35:51	37,4	60,4

DATOS DIA 12		
Hora	Temperatura	Humedad
03:35:51	37,5	60,2
04:35:51	37,5	57,6
05:35:51	37,4	57,8
06:35:51	37,4	59,8
07:35:51	37,8	57,3
08:35:51	37,4	56,5
09:35:51	37,3	59,1
10:35:51	37,4	60,3
11:35:51	37,3	59,3
12:35:51	37,5	55,5
13:35:51	37,6	58,6
14:35:51	37,5	57,8
15:35:51	37,3	59,7
16:35:51	37,8	56,4
17:35:51	37,5	60,5
18:35:51	37,3	57,1
19:35:51	37,5	56,7
20:35:51	37,8	59,1
21:35:51	37,1	47,1
22:35:51	37,4	58,8
23:35:51	37,6	57,2
00:35:51	37,4	59,3
01:35:51	37,6	56,8
02:35:51	37,4	57,8

DATOS DIA 13		
Hora	Temperatura	Humedad
03:35:51	37,5	56,8
04:35:51	37,4	60,1
05:35:51	37,5	58,6
06:35:51	37,8	54,9
07:35:51	37,4	56,1
08:35:51	37,8	58,7
09:35:51	37,3	55,5
10:35:51	37,7	55,7
11:35:51	37,9	56,2
12:35:51	37,4	55
13:35:51	37,8	57,1
14:35:51	37,6	57,4
15:35:51	37,4	59,3
16:35:51	37,6	55,2
17:35:51	37,8	55,9
18:35:51	37,6	59,3
19:35:51	37,4	56,3
20:35:51	37,8	56,7
21:35:51	37,3	57,7
22:35:51	37,7	57,9
23:35:51	37,6	58,9
00:35:51	37,4	57
01:35:51	37,5	56,6
02:35:51	37,4	58,9

DATOS DIA 14		
Hora	Temperatura	Humedad
03:35:51	37,4	59
04:35:51	37,9	58
05:35:51	37,3	57,5
06:35:51	37,5	57,8
07:35:51	37,6	59,6
08:35:51	37,4	58,5
09:35:51	37,1	57,4
10:35:51	37,6	56,2
11:35:51	37,8	57,2
12:35:51	37,5	58,7
13:35:51	37,6	57,6
14:35:51	37,7	57,5
15:35:51	37,2	56,4
16:35:51	37,3	59
17:35:51	37,6	56,5
18:35:51	37,9	58,8
19:35:51	37,3	57,5
20:35:51	37,4	58,7
21:35:51	37,6	58,8
22:35:51	37,8	60,2
23:35:51	37,5	56,5
00:35:51	37,4	56,7
01:35:51	37,5	57,5
02:35:51	37,4	56,9

DATOS DIA 18		
Hora	Temperatura	Humedad
03:35:51	37,5	59,2
04:35:51	37,4	58
05:35:51	37,5	56,9
06:35:51	37,6	56,8
07:35:51	37,6	58,4
08:35:51	37,5	59,2
09:35:51	37,1	50,5
10:35:51	37,8	54,5
11:35:51	37,1	58,8
12:35:51	37,6	58,2
13:35:51	37,4	57,3
14:35:51	37,2	57,8
15:35:51	37,4	56,8
16:35:51	37,9	57,2
17:35:51	37,6	56,6
18:35:51	37,4	58,9
19:35:51	37,3	58,5
20:35:51	37,5	54,2
21:35:51	37,4	58
22:35:51	37,4	56,4
23:35:51	37,4	58,8
00:35:51	37,4	57,3
01:35:51	37,5	59,2
02:35:51	37,4	60

DATOS DIA 19		
Hora	Temperatura	Humedad
03:35:51	36,5	57,4
04:35:51	36,4	58,6
05:35:51	36,5	56,7
06:35:51	36,6	57,7
07:35:51	36,7	59,7
08:35:51	36,6	64,1
09:35:51	36,4	60,5
10:35:51	37	61,8
11:35:51	36,5	62,6
12:35:51	36,1	59
13:35:51	36,8	61,5
14:35:51	36,8	64,8
15:35:51	36,3	60,9
16:35:51	37,6	65,1
17:35:51	36,8	64,7
18:35:51	36,9	64
19:35:51	36,6	63,5
20:35:51	36,6	65,3
21:35:51	36,5	63,6
22:35:51	36,6	63,2
23:35:51	36,7	62,5
00:35:51	36,7	64,7
01:35:51	36,7	65,5
02:35:51	36,7	62,7

DATOS DIA 20		
Hora	Temperatura	Humedad
03:35:51	36,6	60
04:35:51	36,7	62,8
05:35:51	36,5	63,6
06:35:51	36,9	64,8
07:35:51	36,9	65,4
08:35:51	36,6	60,9
09:35:51	36,4	62,3
10:35:51	36,8	62,6
11:35:51	36,8	65,1
12:35:51	36,5	65
13:35:51	36,3	61,6
14:35:51	36,9	64,9
15:35:51	36,7	64,4
16:35:51	37	65,4
17:35:51	36,8	65,9
18:35:51	36,7	64,7
19:35:51	36,7	62,8
20:35:51	36,5	64,9
21:35:51	36,7	61,7
22:35:51	36,7	63,9
23:35:51	36,8	64,3
00:35:51	36,7	65,4
01:35:51	36,7	62,4
02:35:51	36,7	63,2

DATOS DIA 15		
Hora	Temperatura	Humedad
03:35:51	37,7	58,5
04:35:51	37,6	54,8
05:35:51	37,4	57,7
06:35:51	37,9	57,9
07:35:51	37,5	56,8
08:35:51	37,5	56,9
09:35:51	37,1	56,1
10:35:51	37,6	60,3
11:35:51	37,8	56,1
12:35:51	37,8	41
13:35:51	37,5	42,9
14:35:51	37,3	55,7
15:35:51	37,4	58,5
16:35:51	37,6	56,2
17:35:51	37,9	57,6
18:35:51	37,4	58,5
19:35:51	37,1	57,1
20:35:51	37,6	58,5
21:35:51	37,1	58,3
22:35:51	37,9	60,4
23:35:51	37,1	59,1
00:35:51	37,3	60,7
01:35:51	37,1	59,5
02:35:51	37,1	58,2

DATOS DIA 16		
Hora	Temperatura	Humedad
03:35:51	37,3	56,6
04:35:51	37	59,9
05:35:51	37	60,3
06:35:51	37,9	58,8
07:35:51	37,2	60,4
08:35:51	37,4	58,5
09:35:51	37,6	57,6
10:35:51	37,7	58
11:35:51	37,6	57,8
12:35:51	37,6	56,9
13:35:51	37,4	57,8
14:35:51	37,4	56,4
15:35:51	37,2	56,7
16:35:51	37,5	58,4
17:35:51	37,2	56,2
18:35:51	37,4	59,5
19:35:51	37,5	59,6
20:35:51	37,9	56,6
21:35:51	37,6	57
22:35:51	37,5	57,5
23:35:51	37,7	58,3
00:35:51	37,4	57,9
01:35:51	37,5	56
02:35:51	37,4	59,2

DATOS DIA 17		
Hora	Temperatura	Humedad
03:35:51	37,9	57,5
04:35:51	37,3	59,6
05:35:51	37,5	56,3
06:35:51	37,3	56,4
07:35:51	37,1	58,1
08:35:51	37,5	59,4
09:35:51	37,8	58,9
10:35:51	37,3	59,2
11:35:51	37,4	58,6
12:35:51	37,8	55,6
13:35:51	37,5	57,5
14:35:51	37,6	58,6
15:35:51	37,4	58,3
16:35:51	37,1	56,1
17:35:51	37,7	58,5
18:35:51	37	57,5
19:35:51	37,8	57,2
20:35:51	37,5	57,8
21:35:51	37,4	59,9
22:35:51	37,3	57,1
23:35:51	37,9	56,4
00:35:51	37,3	60,2
01:35:51	37,6	57,2
02:35:51	37,4	57,9

DATOS DIA 21		
Hora	Temperatura	Humedad
03:35:51	36,7	63,7
04:35:51	36,7	65,3
05:35:51	36,6	65,3
06:35:51	36,6	63,1
07:35:51	36,7	64,2
08:35:51	36,7	62,8
09:35:51	36,3	63,2
10:35:51	36,7	64,1
11:35:51	36,8	62,7
12:35:51	36,9	63,5
13:35:51	36,7	63,3
14:35:51	36,8	64
15:35:51	36,4	65,9
16:35:51	36,6	61,2
17:35:51	36,6	64,7
18:35:51	36,6	62,8
19:35:51	36,7	61,8
20:35:51	36,7	65,1
21:35:51	36,6	62,7
22:35:51	36,7	62,6
23:35:51	36,4	65
00:35:51	36,7	64,2
01:35:51	36,7	63
02:35:51	36,7	65,7

Anexo 3. Resultados de los procesos de incubación

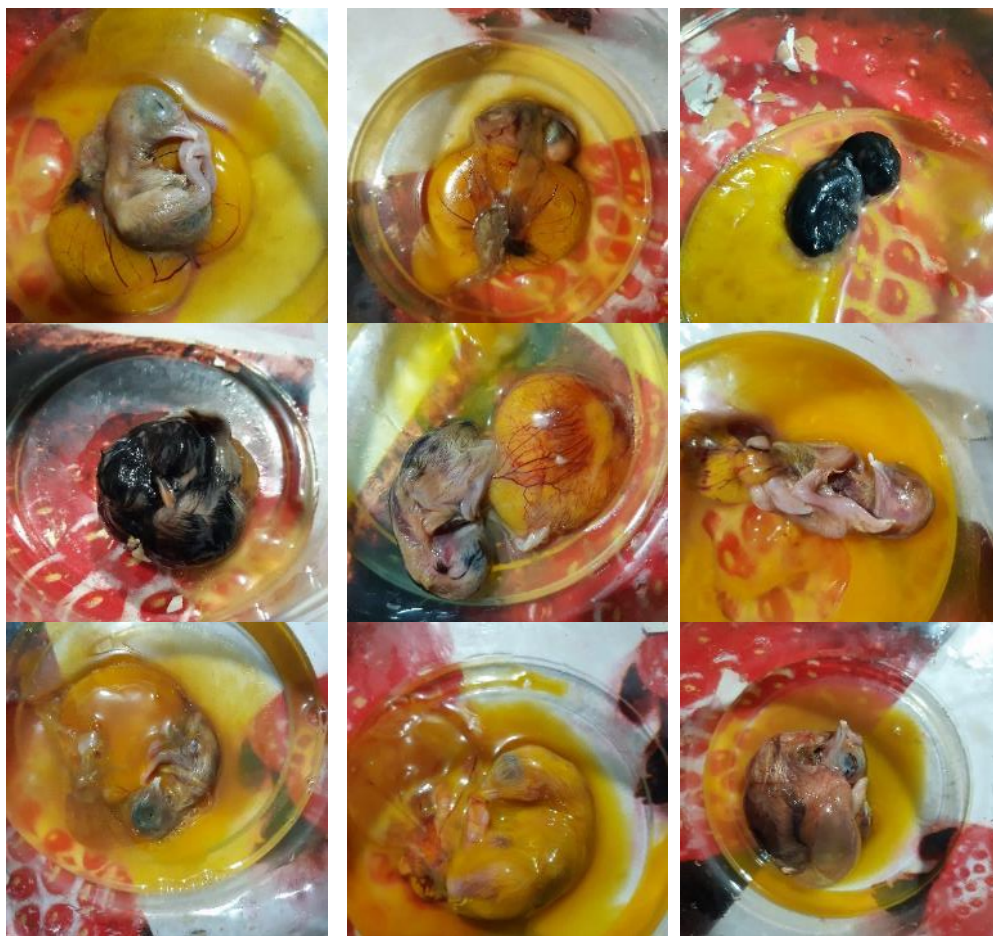


Figura 1-A. Resultados Incubadora HDD mini 56S

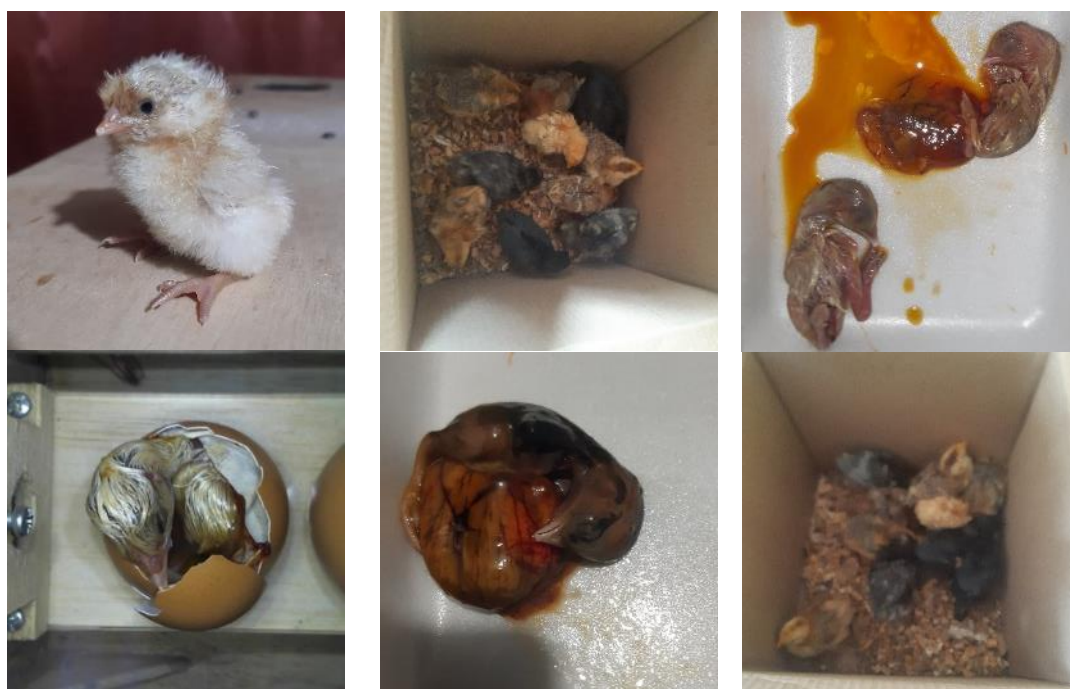


Figura 2-A. Resultado prototipo implementado.

Anexo 4. Pruebas de Ovoscopia



Figura 53. Ovoscopia realizada a los huevos de la incubadora HDD mini 56S

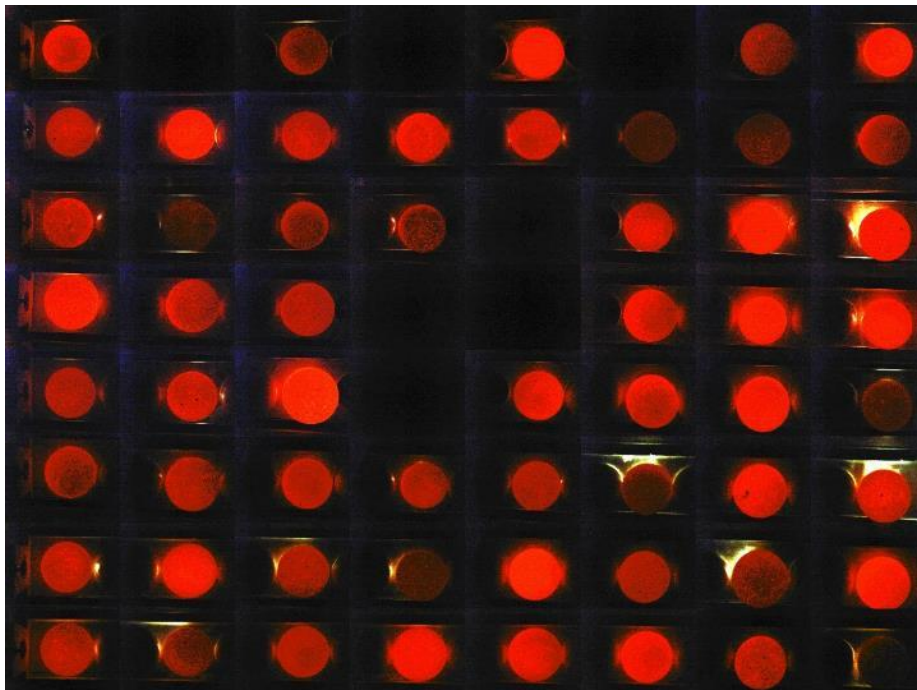
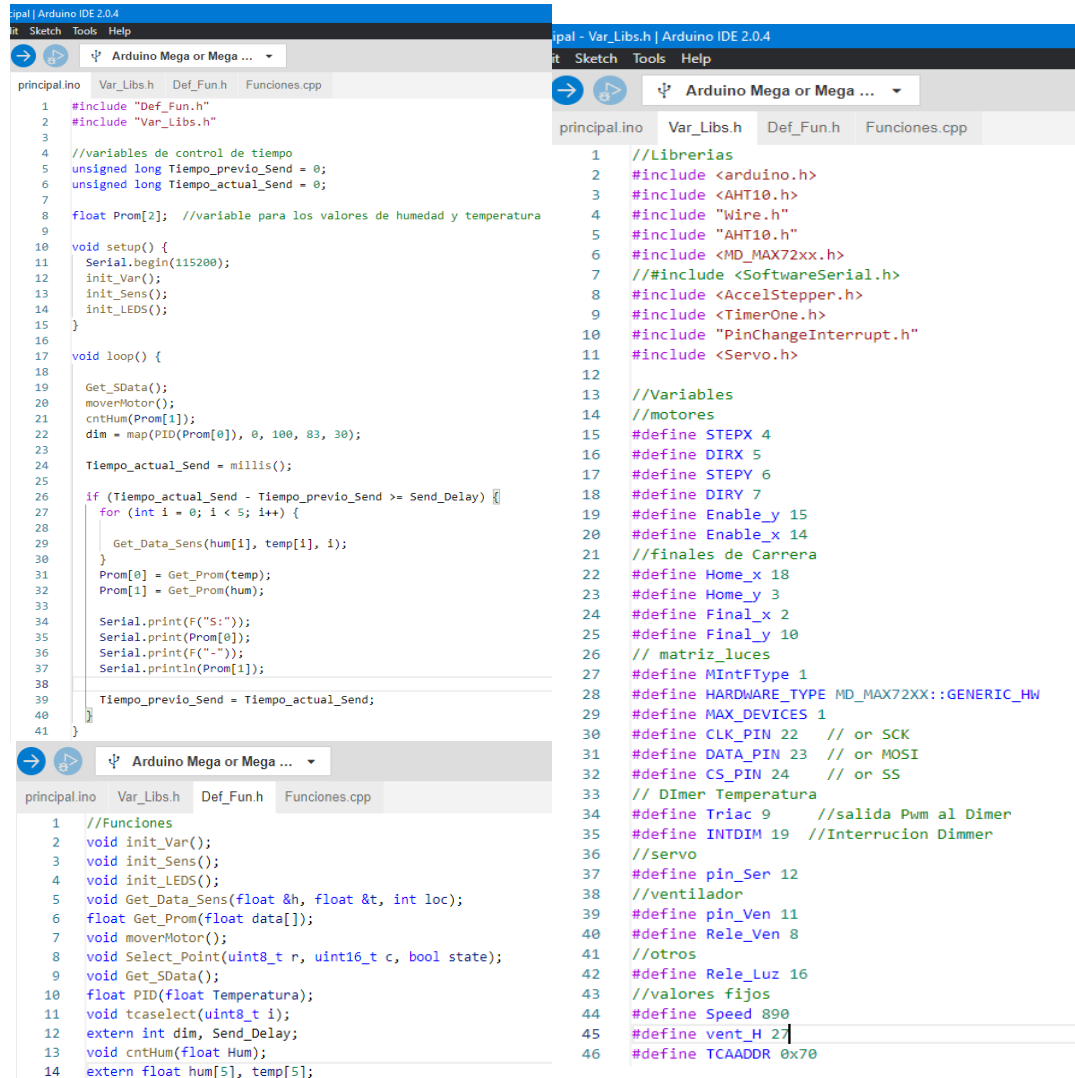


Figura 54. Ovoscopia realizada a los huevos del prototipo.

Anexo 5. Programación en Arduino

Para la programación del Arduino Mega se realizó la programación por pestañas para poder ordenar de mejor manera esta.



```
principal.ino
1 #include "Def_Fun.h"
2 #include "Var_Libs.h"
3
4 //variables de control de tiempo
5 unsigned long Tiempo_previo_Send = 0;
6 unsigned long Tiempo_actual_Send = 0;
7
8 float Prom[2]; //variable para los valores de humedad y temperatura
9
10 void setup() {
11   Serial.begin(115200);
12   init_Var();
13   init_Sens();
14   init_LEDS();
15 }
16
17 void loop() {
18
19   Get_SData();
20   moverMotor();
21   cntHum(Prom[1]);
22   dim = map(PID(Prom[0]), 0, 100, 83, 30);
23
24   Tiempo_actual_Send = millis();
25
26   if (Tiempo_actual_Send - Tiempo_previo_Send >= Send_Delay) {
27     for (int i = 0; i < 5; i++) {
28       Get_Data_Sens(hum[i], temp[i], 1);
29     }
30     Prom[0] = Get_Prom(temp);
31     Prom[1] = Get_Prom(hum);
32
33     Serial.print(F("S:"));
34     Serial.print(Prom[0]);
35     Serial.print(F(" "));
36     Serial.println(Prom[1]);
37
38     Tiempo_previo_Send = Tiempo_actual_Send;
39   }
40 }
41 }
```

```
principal.ino
1 //Librerias
2 #include <arduino.h>
3 #include <AHT10.h>
4 #include "Wire.h"
5 #include "AHT10.h"
6 #include <MD_MAX72xx.h>
7 // #include <SoftwareSerial.h>
8 #include <AccelStepper.h>
9 #include <TimerOne.h>
10 #include "PinChangeInterrupt.h"
11 #include <Servo.h>
12
13 //Variables
14 //motores
15 #define STEPX 4
16 #define DIRX 5
17 #define STEPY 6
18 #define DIRY 7
19 #define Enable_y 15
20 #define Enable_x 14
21 //finales de Carrera
22 #define Home_x 18
23 #define Home_y 3
24 #define Final_x 2
25 #define Final_y 10
26 // matriz_luces
27 #define MIntFType 1
28 #define HARDWARE_TYPE MD_MAX72XX::GENERIC_HW
29 #define MAX_DEVICES 1
30 #define CLK_PIN 22 // or SCK
31 #define DATA_PIN 23 // or MOSI
32 #define CS_PIN 24 // or SS
33 // Dimer Temperatura
34 #define Triac 9 //salida Pwm al Dimer
35 #define INTDIM 19 //Interruccion Dimmer
36 //servo
37 #define pin_Ser 12
38 //ventilador
39 #define pin_Ven 11
40 #define Rele_Ven 8
41 //otros
42 #define Rele_Luz 16
43 //valores fijos
44 #define Speed 890
45 #define vent_H 27
46 #define TCAADDR 0x70
```

```
principal.ino
1 //Funciones
2 void init_Var();
3 void init_Sens();
4 void init_LEDS();
5 void Get_Data_Sens(float &h, float &t, int loc);
6 float Get_Prom(float data[]);
7 void moverMotor();
8 void Select_Point(uint8_t r, uint16_t c, bool state);
9 void Get_SData();
10 float PID(float Temperatura);
11 void tcselect(uint8_t i);
12 extern int dim, Send_Delay;
13 void cntHum(float Hum);
14 extern float hum[5], temp[5];
```

```

Funciones.cpp | Arduino IDE 2.0.4
Funciones.cpp | Arduino IDE 2.0.4
ch Tools Help
ch Tools Help
Arduino Mega or Mega ...
Arduino Mega or Mega ...
ial.ino Var_Libs.h Def_Fun.h Funciones.cpp
ial.ino Var_Libs.h Def_Fun.h Funciones.cpp

#include "arduino.h"
#include "Var_Libs.h"

//Configuración del Modulo de Matriz led
MD_MAX72XX mx = MD_MAX72XX(HARDWARE_TYPE, DATA_PIN, CLK_PIN, CS_PIN, 1);
//Configuración del sensor AHT10
AHT10 myAHT10(0x38);
//asignación de variables para control temporal
unsigned long Actual = 0;
unsigned long Tmp_Prev = 0;
//configuración de variables para la libreria AccelStepper
AccelStepper stepperx = AccelStepper(MIntFType, STEPX, DIRX);
AccelStepper steppery = AccelStepper(MIntFType, STEPY, DIRY);
//configuración de la variable para el servomotor
Servo MyServo;
//variables sensor
float hum[5], temp[5];

//pid variables
unsigned long T_now = 0;
unsigned long T_after = 0;
float setPoint = 37.7;
float setPointHum = 63;
//constantes PID
float Kc = 7;
float Tao_I = 650;

int Time_0;
int Read_Delay = 1000;
float sp = 0;
float PID_error = 0;
float previous_error = 0;
float PID_value = 0;
float Error_INT = 0;
float Potencia = 0;
int valor = 0;
//variables Dimer
volatile int cnt = 0;
volatile boolean cruce_cero = 0;
int dim = 83, Send_Delay = 400;
int T_int = 100;
bool D_state = HIGH;

int i = 0;
int j = 0;

7 //variables de control de los finales de carrera
8 bool cntFY = false;
9 bool cntHY = false;
0 bool cntFX = false;
1 int switchState = 0;
2 int lastSwitchState = 1;
3
4 //variables de distancia para motores
5 long dist_x = 0;
6 long dist_y = 0;
7 long pos_Hx = 0;
8 long pos_Hy = 0;
9 long receivedMMdistance = 0;
0 long receivedDelay = 0; //retardo entre dos pa
1 char receivedCommand;
2 bool newData = false;
3 int pos = 0;
4 //variables para el uso de la matriz led
5 uint8_t row = 0;
6 uint16_t colum = 0;
7 bool state = 0;
8
9 //funciones
0 void stopMotor_Y();
1 void stopMotor_FY();
2 void stopMotor_X();
3 void stopMotor_FX();
4 void zero_cross_detect();
5 void Dimer();
6
7 void tcselect(uint8_t i) {
8     if (i > 7) return;
9
10    Wire.beginTransmission(TCAADDR);
11    Wire.write(i << i);
12    Wire.endTransmission();
13 }
14
15 void init_Var() {
16     //inicializacion Pines
17     pinMode(Triac, OUTPUT); //dimer
18     //motores
19     pinMode(STEPX, OUTPUT);
20     pinMode(STEPY, OUTPUT);
21     pinMode(DIRX, OUTPUT);
22     pinMode(DIRY, OUTPUT);
}

```

```

ial.ino Var_Libs.h Def_Fun.h Funciones.cpp
3 pinMode(Enable_y, OUTPUT);
4 pinMode(Enable_x, OUTPUT);
5 //finales de carrera
6 pinMode(Final_y, INPUT);
7 pinMode(Final_x, INPUT);
8 pinMode(Home_y, INPUT);
9 pinMode(Home_x, INPUT);
0 // otros
1 pinMode(pin_Ven, OUTPUT);
2 pinMode(Rele_Ven, OUTPUT);
3 pinMode(Rele_Luz, OUTPUT);
4 //ventilador
5 pinMode(vent_H, OUTPUT);
6 //Servo
7 MyServo.attach(pin_Ser);
8 MyServo.write(70);
9 //interrupcion Dimer
0 attachInterrupt(digitalPinToInterrupt(INTDIM), zero_cross_detect, R
1 Timer1.initialize(T_int);
2 Timer1.attachInterrupt(Dimer);
3 //interrupciones finales de carrera
4 //attachInterrupt(digitalPinToInterrupt(Home_x), stopMotor_X, RISING
5 attachInterrupt(digitalPinToInterrupt(Final_x), stopMotor_FX, RISING
6 attachInterrupt(digitalPinToInterrupt(Home_y), stopMotor_Y, RISING
7 attachPCINT(digitalPinToPCINT(Final_y), stopMotor_FY, CHANGE);
8 //inicializacion motores
9 stepperx.setMaxSpeed(2000);
0 stepperx.setAcceleration(1000);
1 stepperx.disableOutputs();
2 steppery.setMaxSpeed(2000);
3 steppery.setAcceleration(1000);
4 steppery.disableOutputs();
5 digitalWrite(Enable_y, LOW);
6 digitalWrite(Enable_x, LOW);
7 digitalWrite(Rele_Luz, LOW);
8 digitalWrite(Rele_Ven, HIGH);
9 analogWrite(pin_Ven, 255);
0
1
2 //funcion para inicializar los sensores AHT10
3 void init_Sens() {
4     Wire.begin();
5     for (int i = 0; i <= 4; i++) {
6         tcselect(i);
7         while (myAHT10.begin() != true) {
8             Serial.print("AHT10_");
}

```

```

Serial.println(" no esta conectado o falla al cargar el coeficiente de calibracion
}
}
}
//Funcion para inicializar la matriz led
void init_LEDS() {
    mx.begin();
    mx.control(MD_MAX72XX::INTENSITY, 15);
    mx.clear();
}
//Funcion para detectar el cruce por cero
void zero_cross_detect() {
    cruce_cero = true;
    cnt = 0;
    digitalWrite(Triac, LOW);
}
//Funcion para controlar la potencia de salida del Dimer
void Dimer() {
    if (cruce_cero == true) {
        if (cnt >= dim) {
            digitalWrite(Triac, HIGH);
        } else {
            cnt++;
        }
    }
}
//Funcion para realizar el movimiento de los motores
void moverMotor() {
    if (digitalRead(Final_y) == HIGH && cntFY == false) {
        steppery.setCurrentPosition(61800);
        cntFY = true;
    }
    if (digitalRead(Final_x) == HIGH && cntFX == false) {
        stepperx.setCurrentPosition(40000);
        cntFX = true;
    }
    if (digitalRead(Home_y) == HIGH) {
        dist_y = dist_y;
        detachInterrupt(digitalPinToInterrupt(Home_y));
    }
}

```



```

principalino Var_libs Def_Funh Funciones.cpp
185 if (abs(steppers.currentPosition()) != dist_x) {
186 digitalWrite(Enable_x, LOW);
187 steppers.enableOutputs();
188 steppers.runSpeedToPosition();
189 } else {
190 digitalWrite(Enable_x, HIGH);
191 steppers.disableOutputs(); // disable power
192 }
193 if (abs(steppers.currentPosition()) != dist_y) {
194 long posM = dist_y;
195 if (abs(steppers.currentPosition()) == 1000) {
196 attachInterrupt(digitalPinToInterrupt(More_y), stopMotor_Y, RISING);
197 steppers.setCurrentPosition(1000);
198 dist_y = posM;
199 steppers.moveTo(dist_y);
200 steppers.setSpeed(Speed);
201 }
202 }
203 digitalWrite(Enable_y, LOW);
204 steppers.enableOutputs();
205 steppers.runSpeedToPosition();
206 } else {
207 digitalWrite(Enable_y, HIGH);
208 steppers.disableOutputs(); // disable power
209 }
210 if (abs(steppers.currentPosition()) == dist_x && abs(steppers.currentPosition()) == dist_y) {
211 Send_Delay = 400;
212 }
213 }
214 //Funcion para controlar el encendido/apagado de la matriz led
215 void Select_Point(uint8_t r, uint8_t c, bool state) {
216 mx.setpoint(r, c, state);
217 }
218 }
219 //Funcion para leer datos del puerto serial
220 void get_sData() {
221 if (Serial.available() > 0) {
222 receivedCommand = Serial.read();
223 newData = true;
224 }
225 if (readData == true) {
226 Send_Delay = 10000;
227 if (receivedCommand == 'a') //rover motores a la posicion (x,y) deseada
228 {
229 dist_x = Serial.parseFloat();
230 dist_y = Serial.parseFloat();
231 }
232 if (receivedCommand == 'h') { //retorno a home
233 dist_x = 0;
234 dist_y = 0;
235 steppers.moveTo(float(pos_Mx));
236 steppers.setSpeed(Speed);
237 steppers.moveTo(float(pos_My));
238 steppers.setSpeed(Speed);
239 }
240 if (receivedCommand == 'p') { //obtener posicion
241 Serial.print("D:");
242 Serial.print(steppers.currentPosition());
243 Serial.print("\n");
244 Serial.println(steppers.currentPosition());
245 }
246 if (receivedCommand == 'g') { //control matriz led
247 row = Serial.parseFloat();
248 colom = Serial.parseFloat();
249 state = Serial.parseFloat();
250 Select_Point(row, colom, state);
251 }
252 if (receivedCommand == 's') { //rover servomotor
253 int pos = Serial.parseInt();
254 MyServo.write(pos);
255 }
256 if (receivedCommand == 'd') { //valor calefactor
257 int pos = Serial.parseInt();
258 din = pos;
259 }
260 if (receivedCommand == 'u') { //envio variables de sensores
261 int sen = Serial.parseInt();
262 if (sen != 0) {
263 Serial.print("U:");
264 Serial.print(temp[0], 2);
265 Serial.print("\n");

```

```

principalino Var_libs Def_Funh Funciones.cpp
277 Serial.print(temp[1], 2);
278 Serial.print("\n");
279 Serial.print(temp[2], 2);
280 Serial.print("\n");
281 Serial.println(temp[3], 2);
282 } else {
283 Serial.print("H:");
284 Serial.print(hum[0], 2);
285 Serial.print("\n");
286 Serial.print(hum[1], 2);
287 Serial.print("\n");
288 Serial.print(hum[2], 2);
289 Serial.print("\n");
290 Serial.println(hum[3], 2);
291 }
292 }
293 if (receivedCommand == 'o') { //encender luz camara
294 int com = Serial.parseInt();
295 if (com != 0) {
296 digitalWrite(Rele_Luz, HIGH);
297 } else {
298 digitalWrite(Rele_Luz, LOW);
299 }
300 }
301 if (receivedCommand == 'v') { //encender ventilacion
302 int com = Serial.parseInt();
303 if (com != 0) {
304 digitalWrite(Rele_Ven, HIGH);
305 } else {
306 digitalWrite(Rele_Ven, LOW);
307 }
308 }
309 if (receivedCommand == 'n') { //encender ventilacion humidificador
310 int com = Serial.parseInt();
311 if (com != 0) {
312 digitalWrite(vent_H, HIGH);
313 } else {
314 digitalWrite(vent_H, LOW);
315 }
316 }
317 if (receivedCommand == 'l') { //fijar setpoint temperatura
318 float com = Serial.parseFloat();
319 analogWrite(pin_Ven, com);
320 }
321 if (receivedCommand == 'b') { //setpoint humedad
322 setPointHum = Serial.parseInt();
323 }
324 if (receivedCommand == 't') { //temperatura-humedad sensor 5
325 Serial.print("T:");
326 Serial.print(temp[0], 2);
327 Serial.print("\n");
328 Serial.println(hum[0], 2);
329 }
330 newData = false;
331 }
332 }
333 //funcion del Controlador PID
334 float PID(float Temperatura) {
335 T_nov = millis();
336 //Temperatura = 5.0 * 100.0 * analogRead(A0) / 1024.0;
337 if (T_nov - T_after >= Read_Delay) {
338 T_after += Read_Delay;
339 if (T_nov <= Time_0) {
340 PID_value = 0;
341 } else if (T_nov >= Time_0) {
342 PID_error = setPoint - Temperatura; //Calculo del error
343 Error_INT = Error_INT + PID_error * (1000 / Read_Delay); //Calculo de la integral del error
344 PID_value = Kc * (PID_error + (I / Tao_I) * Error_INT); //Calculo de la salida del controlador PID
345 sp = setPoint;
346 }
347 // Limite de salida del controlador
348 if (PID_value < 0) { PID_value = 0; }
349 if (PID_value > 100) { PID_value = 100; }
350 return (Potencia - PID_value); //Asignacion a la entrada de la planta.
351 }
352 }
353 //Funciones de Finales de Carrera
354 void stopMotor_Y() {
355 detachInterrupt(digitalPinToInterrupt(More_y));
356 steppers.stop(); //stop motor
357 steppers.disableOutputs();
358 digitalWrite(Enable_y, HIGH);
359 steppers.setCurrentPosition(pos_My);
360 Serial.println("Parada inicio motor y");
361 dist_y = 0;
362 }

```

```

principal - Funciones.cpp | Arduino IDE 2.0.4
File Edit Sketch Tools Help

Arduino Mega or Mega ...

principal.ino  Var_Libs.h  Def_Fun.h  Funciones.cpp

370
371 void stopMotor_FY() {
372     switchState = !digitalRead(Final_y);
373     if (switchState != lastSwitchState) {
374         if (switchState == LOW) {
375             steppery.stop(); //stop motor
376             steppery.disableOutputs();
377             digitalWrite(Enable_y, HIGH);
378             dist_y = 61800;
379             steppery.setCurrentPosition(dist_y);
380             // Serial.println("Parada Final motor y");
381         }
382         lastSwitchState = switchState;
383     }
384 }
385
386 void stopMotor_X() {
387     //si motor es true, es el motor x caso contrario es el motor y
388     Serial.println("Parada inicio motor x");
389     stepperx.stop(); //stop motor
390     stepperx.disableOutputs();
391     digitalWrite(Enable_x, HIGH);
392     stepperx.setCurrentPosition(pos_Hx);
393     dist_x = 0;
394 }
395
396 void stopMotor_FX() {
397     // Serial.println("Parada final motor x");
398     stepperx.stop(); //stop motor
399     stepperx.disableOutputs();
400     digitalWrite(Enable_x, HIGH);
401     dist_x = 40000;
402     stepperx.setCurrentPosition(dist_x);
403 }
404
405 //Funcion para obtener los valores de humedad y temperatura
406 void Get_Data_Sens(float &h, float &t, int loc) {
407     tcselect(loc);
408     t = myAHT10.readTemperature();
409     h = myAHT10.readHumidity();
410 }
411
412 //Funcion para generar los valores promedio de temperatura y humedad
413 float Get_Prom(float data[]) {
414     float sensDist[5] = { 1, 3, 1, 3, 20 };
415     data[1] -= 1.0;
416     float total_we = 0;
417     float weiSum = 0;
418     for (int i = 0; i < 5; i++) {
419         float weight = 1.0 / sensDist[i];
420         weiSum += weight * data[i];
421         total_we += weight;
422     }
423     return (weiSum / total_we);
424 }
425
426 //Funcion de Control de Humedad
427 void cntHum(float Hum) {
428     if (Hum < setPointHum) { //55
429         digitalWrite(vent_H, HIGH);
430     } else {
431         digitalWrite(vent_H, LOW);
432     }
433 }

```

Anexo 6. Programación en Python

Para la programación de la Raspberry Pi se optó de igual manera en generar archivos para mantener un control de la programación más ordenado y eficiente.

```

3 Window.borderless = True
1 Window.fullscreen="auto"
2
3 nEgg= [[0, 1, 2, 3, 4, 5, 6, 7],
1      [15, 14, 13, 12, 11, 10, 9, 8],
5      [16, 17, 18, 19, 20, 21, 22, 23],
7      [31, 30, 29, 28, 27, 26, 25, 24],
3      [32, 33, 34, 35, 36, 37, 38, 39],
5      [47, 46, 45, 44, 43, 42, 41, 40],
3      [48, 49, 50, 51, 52, 53, 54, 55],
1      [63, 62, 61, 60, 59, 58, 57, 56]]
2
1
2 Generate tests for the below class
3 class Ui(ScreenManager):
4     pass
5
6 Generate tests for the below class
7 class MainApp(MDApp):
8
9     cred=credentials.Certificate("/home/jomsk/project/cred.
10 counter = NumericProperty(1)
11 ub="/home/jomsk/project/config.ini"
12 dialog=None
13 cntPhoto=0
14 sensor1=0
15 sensor2=0
16 cntMqtt=False
17 prevMqtt=False
18
19 def on_start(self):
20     self.config = configparser.ConfigParser()
21     self.ishome=False
22     self.cam=False
23     self.valinit()
24     self.increment=1
25     t = threading.Thread(target=self.check_Internet)
26     t.daemon = True
27     t.start()
28
29     try:
30         firebase_admin.initialize_app(self.cred,{
31             'databaseURL': 'https://databaseiot-smarth-de
32         })
33     except Exception as e:
34         print("error Conexion a FireBase: ",e)
35
36     self.root.ids.wifi_active=self.check_interface()
37     Clock.schedule_interval(self.update_sensors, 0.5)
38     Clock.schedule_interval(self.DB_ST, 60)
39     self.mqtt.startConnectVR()#internet
40     self.DB.Create_DB()
41     self.init_graph()
42     #Clock.schedule_interval(self.data.set_data,30)
43     self.table()
44     self.Comp.mainApp=self
45     self.data.mainApp=self
46     self.graficas.mainApp=self
47     self.mqtt.mainApp=self
48     self.DB.mainApp=self
49
50 def build(self):
51     self.cam=None
52     self.sensor1=0
53     self.sensor2=0
54     self.initu=0
55     self.net=False
56     self.mqtt=Conexion()
57     self.graficas=Graficas()
58     self.Comp=Incubacion()
59     self.data=Data()
60     self.DB=DB()
61     self.data.start_reading()
62     self.mis_redes=[]
63     self.mis_iden=None
64     self.theme_cls.theme_style="Dark"
65     self.theme_cls.primary_palette="Teal"
66     Builder.load_file('desing.kv')
67     return Ui()
68
69 def check_Internet(self):
70     while True:
71         try:
72             sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
73             # Establecemos un tiempo máximo de espera de 2 segundos
74             sock.settimeout(0.5)
75             # Hacemos la conexión al servidor de prueba
76             sock.connect(("www.youtube.com", 80))
77             self.net= True
78
79 def fun_net(self,*args):
80     if self.net:
81         self.cntMqtt=True
82         self.updateDb()
83     else:
84         self.cntMqtt=False
85
86 def updateDb(self):
87     if self.cntMqtt!=self.prevMqtt:
88         #self.mqtt.startConnectVR()#internet
89         print("ejecucion MQTT")
90         self.prevMqtt=self.cntMqtt
91     try:
92         ref=db.reference('temp')
93         ref.set(self.sensor1)
94         ref=db.reference('hum')
95         ref.set(self.sensor2)
96     except:
97         print('error enviando datos')
98
99 def valinit(self):
100     self.datruta = "/home/jomsk/project/assets/"
101
102     if not os.path.exists(os.path.dirname(self.datruta)):
103         os.makedirs(os.path.dirname(self.datruta+"/imagenes/"))
104         os.makedirs(os.path.dirname(self.datruta+"/files/"))
105         for i in range(4):
106             os.makedirs(os.path.dirname(self.datruta+"/imagenes/imag"+str(i)+"/"))
107         os.makedirs(os.path.dirname(self.datruta+"/system/imagenes/"))
108         os.makedirs(os.path.dirname(self.datruta+"/system/video/"))
109         os.makedirs(os.path.dirname(self.datruta+"/user/imagenes/"))
110         os.makedirs(os.path.dirname(self.datruta+"/user/video/"))
111         for i in range(64):
112             os.makedirs(os.path.dirname(self.datruta+"/system/imagenes/egg"+str(i)+"/"))
113     self.config.read(self.ub)
114     if not self.config.has_section('Posiciones'):
115         self.config.add_section('Posiciones')
116         self.config.set('Posiciones','x','0')
117         self.config.set('Posiciones','y','0')
118
119     else:
120         self.start= int(self.config.get('Incubacion','activa'))
121         self.cntPhoto= int(self.config.get('Incubacion','imag'))
122         self.data.sel_x= int(self.config.get('Posiciones','x'))
123         self.data.sel_y= int(self.config.get('Posiciones','y'))
124         self.data.send_pos("M"+str(Pos_xlist[self.data.sel_x])+" "+str(Pos_ylist[self.data.sel_y]))
125
126         self.config.set('Incubacion','imag','0')
127
128     if not self.config.has_section('Configuracion'):
129         self.config.add_section('Configuracion')
130         self.config.set('Configuracion','hunspl','60')
131         self.config.set('Configuracion','temppl','37')
132         self.config.set('Configuracion','timeturn','2')
133         self.config.set('Configuracion','timeovo','3')
134         self.config.set('Configuracion','getdatetime','1')
135         self.config.set('Configuracion','alhum','50')
136         self.config.set('Configuracion','aitemp','36.5')
137         self.config.set('Configuracion','database','0')
138     with open(self.ub, 'w') as configFile:
139         self.config.write(configFile)
140
141 def check_interface(self):
142     # Obtiene información sobre todas las interfaces de red
143     net_io = psutil.net_if_addrs()
144     # Verifica si la interfaz específica está presente
145     if "wlan0" in net_io:
146         # Verifica si la interfaz está activa o no
147         if net_io["wlan0"][0].family == 2:
148             return True
149         else:
150             return False
151     else:
152         return False
153
154 def check_bluetooth_status(self):
155     result = subprocess.run(["bluetoothctl", 'show'], stdout=subprocess.PIPE)
156     output = result.stdout.decode()
157     if 'Powered: yes' in output:
158         return True
159     else:
160         return False
161
162 def check_pos(self):
163     if self.data.sel_x >= 0:
164         self.root.ids.btn_left.disabled=False
165     else:
166         self.root.ids.btn_left.disabled=True
167     if self.data.sel_y != 0:
168         self.root.ids.btn_down.disabled=False
169     else:
170         self.root.ids.btn_down.disabled=True

```

```

else:
    self.root.ids.btn_right.disabled=True
if self.data.sel_y < 7:
    self.root.ids.btn_up.disabled=False
else:
    self.root.ids.btn_up.disabled=True
if (self.data.sel_y or self.data.sel_x) == 0:
    self.root.ids.btn_home.disabled=True
else:
    self.root.ids.btn_home.disabled=False

def egg_angle(self, volteo):
    if volteo:
        self.root.ids.angle_egg.disabled=False
        if self.counter==2:
            self.increment--1
        elif self.counter==0:
            self.increment=1
        self.counter+=self.increment
        print(str(self.counter))
        if self.counter==2:
            com="s 54"
            self.data.send_com(com)
        elif self.counter==1:
            com="s 79"
            self.data.send_com(com)
        elif self.counter==0:
            com="s 106"
            self.data.send_com(com)
    else:
        self.root.ids.angle_egg.disabled=True

def DB_ST(self,*args):
    self.started=int(self.config.get('Incubacion','activa'))
    if self.started==1:
        self.DB.set_sc_data(
            self.sensor1,
            self.sensor2,
            int(self.config.get('Incubacion','dia'))
        )

def update_label(self,*args):
    self.root.ids.time.text=time.strftime('%H:%M')
    self.data.get_pos()
    self.DB.newTable()
    self.DB.set_data(self.sensor1,self.sensor2)
    self.check_pos()
    if self.started==1:
def reset_max_min(self,*args):
    self.data.reset_max_min()

def updateDay(self):
    self.Comp.Time()
    self.config.read(self.ub)
    self.root.ids.btn_inicio.text="Dia: " + self.config.get('Incubacion','dia')
    if self.Comp.getImage():
        self.intImg()

def startIncu(self):
    self.Comp.inicio()
    self.config.read(self.ub)
    self.root.ids.btn_inicio.text="Dia: " + self.config.get('Incubacion','dia')
    self.root.ids.btn_inicio.disabled=True

def update_sensors(self, dt):
    # Leer datos del puerto serie
    self.sensor1,self.sensor2,sensor1_max,sensor1_min,sensor2_max,sensor2_min=self.data.get_data()
    # Actualizar etiquetas con los valores de sensor1 y sensor2
    self.root.ids.Temp.text=str(self.sensor1)
    self.root.ids.hum.text=str(self.sensor2)
    self.root.ids.Temp.ids.H.text.text=str(sensor1_max)
    self.root.ids.Temp.ids.L.text.text=str(sensor1_min)
    self.root.ids.hum.ids.H.text.text=str(sensor2_max)
    self.root.ids.hum.ids.L.text.text=str(sensor2_min)
    self.root.ids.Temp.ids.value_change.progress=int(round(self.sensor1 / 60 * 270))
    self.root.ids.hum.ids.value_change.progress=int(round(self.sensor2 / 100 * 270))

def initing(self):
    self.data.send_com("o 1")
    self.data.send_com("s 79")
    self.cntPhoto=int(self.config.get('Incubacion','imag'))
    self.image_list=[]
    self.root.ids.camara_main.disabled=True
    self.openCamera()
    Clock.schedule_interval(self.takeIng, 20)
    time.sleep(2)
    if (self.data.posxRl==0 or self.data.posyRl==0) and self.ishome==False:
        self.data.send_data('home')
    else:
        print("esta en home")
        self.takeIng(datetime.datetime.now())

def openCamera(self):
    if self.cap.isOpened():
        self.cap.set(cv2.CAP_PROP_FOCUS, -10)
        self.cap.set(cv2.CAP_PROP_AUTOFOCUS, 1)
        print("Camara Encendida")
        return True
    except cv2.error as e:
        print("Error iniciando la Camara: ",e)
    return False

def takeIng(self,dt):
    if self.cap is None or not self.cap.isOpened():
        if not self.openCamera():
            print("Error al iniciar la Camara")
            time.sleep(0.3)
            return
    if (self.data.posxRl==0 or self.data.posyRl==0) and self.ishome==False:
        self.data.send_data('home')
    else:
        if self.data.sel_y2==0:
            self.rowRlPhoto()
        else:
            self.rowLPhoto()

def prosing(self):
    row_list=[]
    for i,file in enumerate(range(0,64,8)):
        row=np.hstack(self.image_list[1:i+8])
        if i%2==0:
            row_list.append(np.hstack(self.image_list[file:file+8]))
        else:
            row_list.append(np.hstack(self.image_list[file+8][:i+1]))
    row_list.reverse()
    mosaic=np.vstack(row_list)
    cv2.imwrite("/home/josak/project/assets/images/Iragen_Completa "+ str(self.cntPhoto)+".jpg",mosaic)

def rowRlPhoto(self):
    if self.data.posxRl==Pos_xlist[7]:
        print("Foto Tomada")
        ret, self.prev_img=self.cap.read()
        now = datetime.datetime.now()
        flipping=cv2.flip(self.prev_img,1)
        self.image_list.append(flipping)
        filename = "Imagen ({self.cntPhoto}).jpg"
        cv2.imwrite(self.datruta+'system/images/egg ('+str(negg[self.data.sel_y][self.data.sel_x]+)"/'
            + filename, flipping)
        self.data.send_data('right')
        self.ishome=True
        self.cap.release()
        self.data.send_data('up')

def rowLPhoto(self):
    if self.data.posxRl==Pos_xlist[0]:
        print("Foto Tomada")
        ret, self.prev_img=self.cap.read()
        now = datetime.datetime.now()
        flipping=cv2.flip(self.prev_img,1)
        self.image_list.append(flipping)
        filename = "Imagen ({self.cntPhoto}).jpg"
        cv2.imwrite(self.datruta+'system/images/egg ('+str(negg[self.data.sel_y][self.data.sel_x]+)"/'
            + filename, flipping)
        self.data.send_data('left')
        self.ishome=True
        self.cap.release()
    else:
        print("Foto Tomada")
        ret, self.prev_img=self.cap.read()
        now = datetime.datetime.now()
        flipping=cv2.flip(self.prev_img,1)
        self.image_list.append(flipping)
        filename = "Imagen ({self.cntPhoto}).jpg"
        cv2.imwrite(self.datruta+'system/images/egg ('+str(negg[self.data.sel_y][self.data.sel_x]+)"/'
            + filename, flipping)
        self.ishome=True
        self.cap.release()
    if self.data.posxRl==Pos_xlist[0] and self.data.posyRl== Pos_xlist[7]:
        self.data.send_data("home")
        self.data.send_com("o 0")
        self.root.ids.camara_main.disabled=False
        print("Fin Funcion photos")
        self.ishome=False
        self.prosing()
        self.cntPhoto-=1
        self.guardar()
        Clock.unschedule(self.takeIng)
    else:
        self.data.send_data('up')

```



```

17 self.root.ids.user.text== admin and self.root.ids.password.text == admin :
    self.root.current='screen1'
    self.root.ids.conf.disabled=False
    self.root.ids.log.disabled=True
    #Ingresar opciones habilitadas
    self.root.ids.hum.text='86%'#Cambiar valor de medir en numeros
    self.root.ids.hum.ids.H.text.text='90%'##LINEA PARA CAMBIAR VALOR en max y mi
    self.root.ids.hum.ids.value_change.progress=205#Cambiar valor barra de prog
else:
    if not self.dialog:
        self.dialog = MDDialog(
            title="ADVERTENCIA",
            text=f"USUARIO INCORRECTO!",
            radius=[20, 7, 20, 7],
            buttons=[
                MDFlatButton(
                    text="OK",
                    theme_text_color="Custom",
                    text_color=self.theme_cls.primary_color,
                    on_release=lambda _: self.dialog.dismiss()
                ),
            ],
        )
        self.dialog.open()

def reback(self):
    self.root.current='screen1'
    self.root.ids.user.text=''
    self.root.ids.password.text=''
    #Ingresar intruccion para desabilitar opciones de configuracion

def gp_style(self):
    self.fig.set_facecolor((.10,.10,.10,1))
    self.ax.set_facecolor((.10,.10,.10,1))
    self.ax.set_xlabel('HORA',color='white')
    self.ax.set_ylabel('VALOR',color='white')
    self.ax.tick_params(axis='x', colors='white')
    self.ax.tick_params(axis='y', colors='white')
    self.ax.spines['left'].set_color('white')
    self.ax.spines['bottom'].set_color('white')
    # Plot some data on the axes
    plt.grid(color='gray', linestyle='--', linewidth=0.50, alpha=0.5)
    plt.grid(True)
    plt.xticks(rotation=60,color='white')
    plt.yticks(color='white')

def init_graph(self):
485 self.fig, self.ax = plt.subplots(constrained_layout=True)
486 self.gp_style()
487 self.graph = FigureCanvasKivyAgg(self.fig)
488 box=self.root.ids.grap_scn
489 box.add_widget(self.graph)
490
491 def grap_cls(self,**kwargs):
492 box=self.root.ids.grap_scn
493 box.clear_widgets()
494
495 def grap_title(self,value,value1):
496
497 if(value=="Select One" or value1=="Select Day"):
498 self.root.ids.gpr_title.text="Grafico"
499 else:
500 self.root.ids.gpr_title.text=f"{value} Dia {value1}"
501
502 def guardar(self):
503 self.config.read(self.ub)
504 self.config.set('Incubacion','imag',str(self.cntPhoto))
505 self.config.set('Posiciones','x',str(self.data.sel_x))
506 self.config.set('Posiciones','y',str(self.data.sel_y))
507 with open(self.ub, 'w') as configfile:
508 self.config.write(configfile)
509
510 def initCam(self):
511 self.openCamera()
512 #self.capture.set(cv2.CAP_PROP_FRAME_WIDTH, 680)
513 #self.capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 480)
514 self.cap.set(cv2.CAP_PROP_AUTOFOCUS, 1)
515 #self.capture.set(cv2.CAP_PROP_FOCUS, 0.6) # ajustar manualmente el enfoque
516 Clock.schedule_interval(self.camUpdate, 1.0/25.0)
517
518 def stopCam(self):
519 Clock.unschedule(self.camUpdate)
520 self.cap.release()
521
522 def camUpdate(self,dt):
523 # display image from cam in opencv window
524 ret, self.frame = self.cap.read()
525 self.cap.set(cv2.CAP_PROP_FOCUS, -10)
526
527 if self.camF:
528 self.frame=self.frame
529 self.procesuz()
530 else:
531 self.frame=cv2.convertScaleAbs(self.frame,alpha=10,beta=0)
532
533 if ret:
534 buf1 = cv2.flip(self.frame, -1)

```

```

#If working on RASPBERRY PI, use colorfmt='rgba' here instead, but stick with "bgr" in bitl
texture1.blit_buffer(buf, colorfmt='bgr', bufferfmt='ubyte')
img1=self.root.ids.camera
# display image from the texture
img1.texture = texture1
self.guardar()
else:
    ret, self.frame = self.cap.read()

def savePhoto(self):
    now = datetime.datetime.now()
    buf1 = cv2.flip(self.frame, 1)
    current_datetime = now.strftime("%Y%m%d_%H%M%S")
    filename = f"imagen_{current_datetime}.jpg"
    cv2.imwrite("/home/jonski/project/assets/imagenes/" + filename, buf1)

def procesuz(self):
    try:
        self.iframe=cv2.cvtColor(self.frame,alpha=0.55,beta=0)
        gray=cv2.cvtColor(self.iframe,cv2.COLOR_BGR2GRAY)

        blur = cv2.medianBlur(gray,19)
        canny = cv2.Canny(blur, 50, 120)

        cnts,_ = cv2.findContours(canny.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

        # Encontrar el contorno circular más grande
        max_contour = None
        max_contour_area = 0
        for contour in cnts:
            area = cv2.contourArea(contour)
            if area > max_contour_area:
                max_contour = contour
                max_contour_area = area
            if max_contour is not None:
                ((x, y), radius) = cv2.minEnclosingCircle(max_contour)
                #if radius>110 and radius<170:
                #cv2.circle(self.frame, (int(x), int(y)), int(radius), (0, 255, 0), 0)
        except cv2.error as e:
            print("error: ",e)

def add_notf(self,tipo,mensaje):
    # Create a new button
    new_button = MDBoxLayout(
        md_bg_color='gray',
        radius=10,
        spacing=400,
        padding=[25dp, 10dp, 10dp, 10dp],
        icon="alert-decagram",
        theme_text_color="Custom",
        text_color="yellow",
    )
    new_button.add_widget(self.icons)
    self.txt_notf=MDLabel(
        text=f"[tipo]\n[mensaje]",
        theme_text_color="Custom",
        text_color="black",
        text_size="28dp"
    )
    new_button.add_widget(self.txt_notf)
    new_button.add_widget(
        MDIconButton(
            icon="close-circle",
            theme_text_color="Custom",
            text_color="white",
            on_press=self.change_text
        )
    )
    # Add the new button to the ScrollView
    self.root.ids.scroll_view.add_widget(new_button)
    self.initu=1

def change_text(self,instance):
    Item=instance.parent
    index=self.root.ids.scroll_view.children.index(item)
    self.root.ids.scroll_view.remove_widget(item)

def moving(self,dir,id):
    self.mis_iden=id
    self.data.send_data(dir)

def control(self,state,id):
    if id == "luz_can":
        if state!=True:
            com="o 0"
        else:
            com="o 1"
    elif id == "vent":
        if state!=True:
            com="v 0"
        else:
            com="v 0"

```

```

        com="d "+str(state)

self.data.send_com(com)
print(com)

def toggle_button_state(self, instance):
x=self.data.sel_x
y=self.data.sel_y
if instance.icon == "lightbulb":
    self.camF=True
    instance.icon = "lightbulb-on"
    self.data.send_com("o 1")
    self.mqtt.public_main('control/Cled','true')
elif instance.icon=="lightbulb-on":
    instance.icon = "lightbulb"
    self.data.send_com("o 0")
    self.mqtt.public_main('control/Cled','false')
    self.camF=False
elif(instance.icon=="led-off"):
    instance.icon="led-on"
    com="g "+str(x)+" "+str(y)+" 1"
    self.mqtt.public_main('control/Pos',str(nEgg[x][y]))
    self.mqtt.public_main('control/Nled','true')
    self.data.send_com(com)
elif(instance.icon=="led-on"):
    instance.icon="led-off"
    com="g "+str(x)+" "+str(y)+" 0"
    self.mqtt.public_main('control/Pos',str(nEgg[x][y]))
    self.mqtt.public_main('control/Nled','false')
    self.data.send_com(com)

def table(self):
BoxTab=self.root.ids.boxTable
self.data_tables=MDDataTable(
    use_pagination=True,
    rows_num=20,
    check=False,
    widget_style='ios',
    # name column, width column, sorting function column(optional), custom tooltip
    column_data=[
        ("No.", dp(11)),
        ("Fecha", dp(30)),
        ("Hora", dp(25)),
        ("Dia", dp(12)),
        ("Temperatura", dp(25)),
        ("Humedad", dp(20)),
    ]
)
BoxTab.add_widget(self.data_tables)

def show_redes(self):
self.mis_redes.clear()
self.root.ids.md_list.clear_widgets()
self.queue = multiprocessing.Queue()
self.process = multiprocessing.Process(target=self.star_wifi_scan, args=(self.queue,))
self.process.start()
self.mis_redes = self.queue.get()
for var in self.mis_redes:
    self.root.ids.md_list.add_widget(OneLineListItem(text=var))

def star_wifi_scan(self, queue):

output = os.popen("sudo iwlist wlan0 scan | grep ESSID").read()
output = re.findall(r'ESSID:"(.*?)"',output)
output = [x for x in output if x != '']
queue.put(output)

def scan_wifi(self,value):

if value:
    print("wifi on")
    self.root.ids.btn_wifi.disabled=False
    if self.check_interface() == False:
        subprocess.call(["sudo", "ip", "link", "set", "wlan0", "up"])
    else:
        print("wifi off")
        self.net=False
        subprocess.call(["sudo", "ip", "link", "set", "wlan0", "down"])
    self.root.ids.md_list.clear_widgets()
    self.root.ids.btn_wifi.disabled=True

def activate_bth(self,value):
if value:
    if self.check_bluetooth_status() is False:
        subprocess.call(["sudo", "hciconfig", "hci0", "up"])
        print("bth on")
    else:
        if self.check_bluetooth_status() is True:
            subprocess.call(["sudo", "hciconfig", "hci0", "down"])

def close_app(self, instance):
self.guardar()
self.data.stop()
735     MDApp.get_running_app().stop()
736
737     def on_stop(self):
738         self.guardar()
739         self.data.stop()
740         self.mqtt.stop()
741
742         def reiniciar_aplicacion(self):
743             self.guardar()
744             self.data.stop()
745             MDApp.get_running_app().stop()
746
747 if __name__ == "__main__":
748     MainApp().run()

```

```

screen.py 1  config.ini  Comp_incu.py  getData.py  Local_db.py  Mqtt_conexion.py  Gr
Comp_incu.py > ...
Generate tests for the below class
7 class Incubacion():
8     mainApp=None
9     ejecAn=False
10    ejecDs=False
11    config=configparser.ConfigParser()
12    delta=None
13    ub='/home/jomsk/project/config.ini'
14
15    def inicio(self):
16        startIncu=datetime.now()
17        self.config.read(self.ub)
18        if self.config.get('Incubacion','activa') == '0':
19            self.config.set('Incubacion','inicio',str(startIncu))
20            self.config.set('Incubacion','activa','1')
21            with open(self.ub, 'w') as configfile:
22                self.config.write(configfile)
23
24    def Time(self):
25        self.config.read(self.ub)
26        if not self.config.get('Incubacion','inicio') == '0':
27            inicio=datetime.strptime(self.config.get('Incubacion','inicio'),"%Y-%m-%d %H:%M:%S.%f")
28            self.deltaT= datetime.now()-inicio
29            self.config.set('Incubacion','dia',str(self.deltaT.days))
30            with open(self.ub, 'w') as configfile:
31                self.config.write(configfile)
32            if int(self.config.get('Incubacion','dia')) < 19:
33                #set setpoin temp a 37.7 hum a55-60
34                if not self.ejecAn :
35                    Data().send_com('l '+ self.config.get('Configuracion','temp'))
36                    Data().send_com('b '+ self.config.get('Configuracion','humsp'))
37                    self.ejecAn=True
38                if not self.config.get('Incubacion','inicio') == '0':
39                    self.config.read(self.ub)
40                    self.config.set('Configuracion','temp','37.4')
41                    self.config.set('Configuracion','humsp','59.4')
42                    with open(self.ub, 'w') as configfile:
43                        self.config.write(configfile)
44                if (round(((self.deltaT.seconds/3600)%2),3))==2:
45                    self.mainApp.egg_angle(True)
46            else:
47                #set setpoin hum 65 y temp 36.5
48                if not self.ejecDs :
49                    Data().send_com('l '+ self.config.get('Configuracion','temp'))
50                    Data().send_com('b '+ self.config.get('Configuracion','humsp'))
51                    Data().send_com('s 79')
52                    self.mainApp.egg_angle(False)
53                    self.ejecDs=True
54                if not self.config.get('Incubacion','inicio') == '0':
55                    self.config.read(self.ub)
56                    self.config.set('Configuracion','temp','36.7')
57                    self.config.set('Configuracion','humsp','65')
58                    with open(self.ub, 'w') as configfile:
59                        self.config.write(configfile)
60            else:
61                print(self.deltaT)
62
63    def getImage(self):
64        if (datetime.now().strftime("%H:%M:%S")=="09:00:00")or(datetime.now().strftime("%H:%M:%S")=="21:00:00"):#(round(((self.deltaT.seconds/3600)%12),3))--12:
65            return True
66        else:
67            return False
68
69    def notifi(self):
70        try:
71            inic=datetime.strptime(self.config.get('Incubacion','inicio'),"%Y-%m-%d %H:%M:%S.%f")
72            now=datetime.now()
73            hora=now.strftime("%H:%M:%S")
74            delta=inic-now
75            if ((delta.seconds/3600)%1)--0:
76                if self.mainApp.sensor1 < float(self.config.get('Configuracion','altemp')):
77                    self.mainApp.root.ids.noti.text_color="yellow"
78                    self.mainApp.add_Notf("Temperatura",
79                                         str(hora)+"\nAlerta la temperatura se encuentra por debajo de su rango")
80                if self.mainApp.sensor2 < float(self.config.get('Configuracion','alhum')):
81                    self.mainApp.root.ids.noti.text_color="yellow"
82                    self.mainApp.add_Notf("Humedad",
83                                         str(hora)+"\nAlerta la humedad se encuentra por debajo de su rango")
84            except Exception as e:
85                print("error notificacion: ",e)
86

```

Graph_db.py > ...

```
7 class Graficas():
8     mainApp=None
9
10    def Get_Data(self,dia,sensor):
11        self.mainApp.ax.cla()
12        try:
13            tabla_actual = f"Dia ({dia})"
14            nueva_tabla = type(tabla_actual, (Variables,),
15                               {"Meta": type("Meta", (object,)),
16                                {"table_name": tabla_actual}})
17            datos=nueva_tabla.select().order_by(nueva_tabla.id.asc())
18            Labels= [dato.hora.strftime('%H:%M:%S') for dato in datos]
19            x= [dato.id for dato in datos]
20            if sensor=="Temperatura":
21                y= [dato.temperatura for dato in datos]
22            elif sensor=="Humedad":
23                y= [dato.humedad for dato in datos]
24
25
26            self.mainApp.gp_style()
27            print(Labels)
28            print(y)
29
30            plt.xticks(x,Labels)
31            self.mainApp.ax.plot(x,y)
32            self.mainApp.graph.draw()
33        except peewee.OperationalError as e:
34            print(f'la tabla no existe:{e}')
35        except UnboundLocalError as e:
36            print(f'variable sin asignar')
37
38    def Get_TB_Data(self):
39        try:
40            rows=Variables1.select().dicts()
41            result=[list(row.values())for row in rows]
42            self.mainApp.data_tables.row_data=result
43        except (OperationalError, ProgrammingError) as e:
44            print(e)
45
```

htData.py > Data

```
mi_thread_stop = threading.Event()
Pos_xlist=[0,5300,11200,16900,22800,28900,34900,39900]
Pos_ylist=[0,8600,17200,26000,34800,43500,52400,61550]
```

Generate tests for the below class

```
class Data():
    mainApp=None
    try:
        ser = serial.Serial('/dev/ttyACM0', baudrate=115200)
        mi_thread_stop.clear()
    except serial.SerialUtil.SerialException:
        print("No se pudo abrir el puerto serie. Compruebe la conexión y vuelva a intentarlo.")
        mi_thread_stop.set()

    def start_reading(self):
        # Crear hilo para la lectura del puerto serie
        self.sel_x,self.sel_y=0,0
        self.posxR, self.posyR=0,0
        self.sensor1,self.sensor2=0,0
        self.max_s1,self.min_s1=0,0
        self.max_s2,self.min_s2=0,0
        self.push=False
        self.posx,self.posy=0,0
        self.thread = threading.Thread(target=self.read_serial)
        self.thread.start()

    def read_serial(self):
        while not mi_thread_stop.is_set():
            # Leer datos del puerto serie
            try:
                data = self.ser.readline().strip()
                info=data.decode().split(':')
                if len(info)==2:
                    comand=info[0]
                    data_list = info[1].split("-")
                    if str(comand)=="S":
                        self.sensor1, self.sensor2 = [round(float(val), 1) for val in data_list]
                    elif str(comand=="D"):
                        self.posxR, self.posyR = [float(val) for val in data_list]
                    else:
                        print(data)
            except serial.SerialException:
                print("Error: conexión perdida")
                self.ser.close()
                self.ser = serial.Serial('/dev/ttyACM0', baudrate = 115200)

        except Exception as e:
            print("Error desconocido: ",e)
        pass
```

```
if ((self.posxR==0 or self.posyR==0)and (self.push is True)):
    self.ser.write(self.comando)
    self.ser.write(self.comando)
    self.push=False
except Exception as e:
    print("getPos: ",e)
```

```
def get_data(self):
    if self.sensor1 > self.max_s1:
        self.max_s1=self.sensor1
    elif (self.sensor1<self.max_s1) or self.min_s1==0:
        self.min_s1=self.sensor1

    if self.sensor2 > self.max_s2:
        self.max_s2=self.sensor2
    elif (self.sensor2<self.max_s2) or self.min_s2==0:
        self.min_s2=self.sensor2

    return self.sensor1,self.sensor2,self.max_s1,self.min_s1,self.max_s2,self.min_s2
```

```
def getSens(self):
    return self.T_s0,self.T_s1,self.T_s2,self.T_s3
```

```
def reset_max_min(self):
    self.max_s1=0
    self.min_s1=0
```

```
def send_data(self,dir):
    self.push=True
    global Pos_xlist,Pos_ylist
    if dir=="up":
        if self.sel_y<7:
            self.sel_y+=1
        else:
            self.sel_y=7
    elif dir=="down":
        if self.sel_y>0:
            self.sel_y-=1
        else:
            self.sel_y=0
    elif dir=="right":
        if self.sel_x<7:
```



```

108         self.sel_x+=1
109     else:
110         self.sel_x=7
111 elif dir=="left":
112     if self.sel_x>0:
113         self.sel_x-=1
114     else:
115         self.sel_x=0
116 else:
117     self.sel_x,self.sel_y=0,0
118
119 print(str(self.sel_x)+ " "+str(self.sel_y))
120 self.posx,self.posy=Pos_xlist[self.sel_x],Pos_ylist[self.sel_y]
121 self.comando="m "+str(self.posx)+" "+str(self.posy)
122
123 self.comando=self.comando.encode()
124
125 self.ser.write(self.comando)
126
127
128 def send_com(self,data):
129     com=data.encode()
130     try:
131         self.ser.write(com)
132     except Exception as e:
133         print("Error getdata: ")
134
135
136
137 def stop(self):
138     mi_thread_stop.set()
139
140
141 Generate tests for the below class
142 class AuthFB():
143     def getUsers(self):
144         user_list=auth.list_users()
145         users=[]
146         for user in user_list.users:
147             users.append([user.email,user.uid,user.disabled])

```

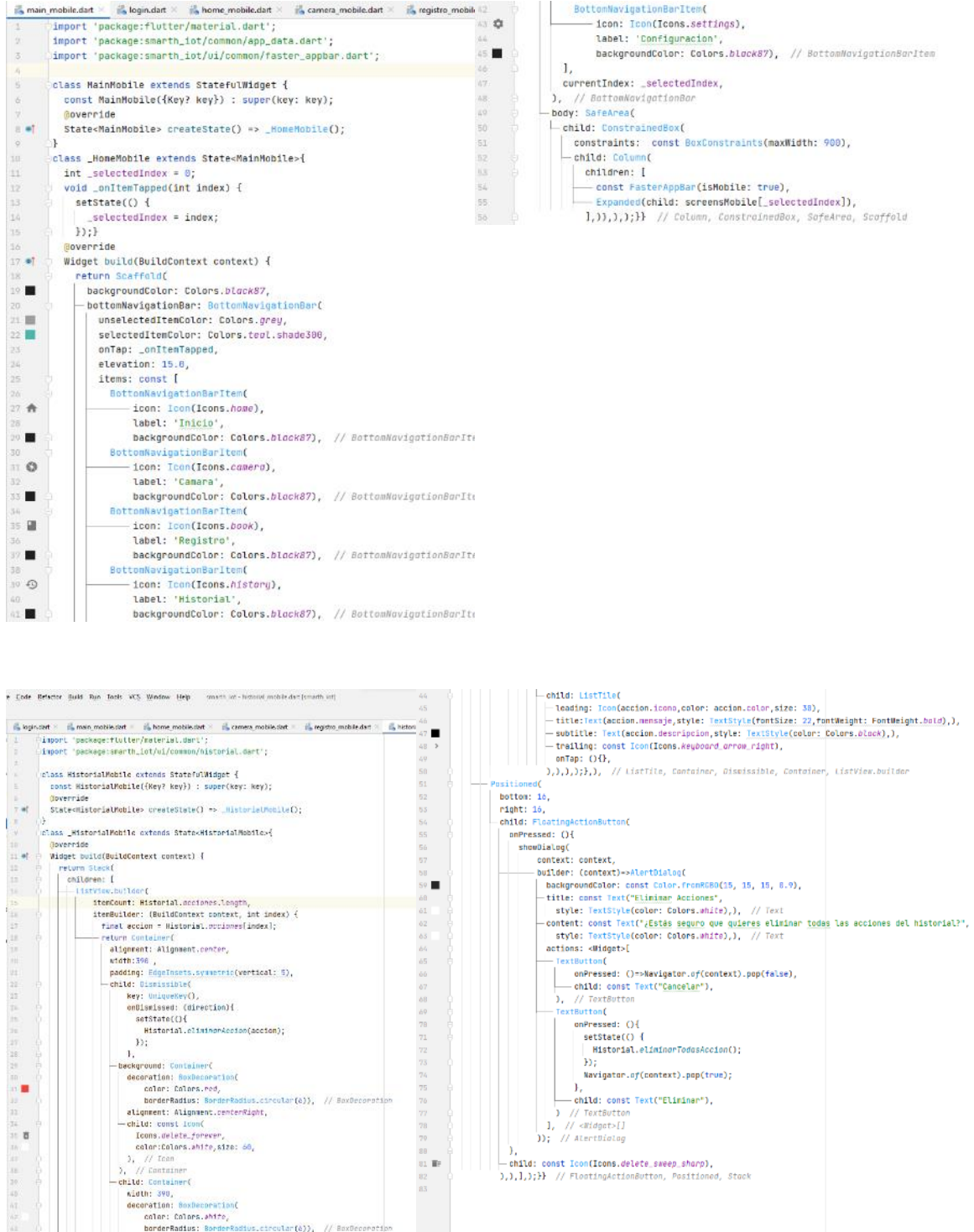
```

7 #! /usr/bin/env python
8 # coding: utf-8
9 #! /usr/bin/env python
10 # coding: utf-8
11 #! /usr/bin/env python
12 # coding: utf-8
13 #! /usr/bin/env python
14 # coding: utf-8
15 #! /usr/bin/env python
16 # coding: utf-8
17 #! /usr/bin/env python
18 # coding: utf-8
19 #! /usr/bin/env python
20 # coding: utf-8
21 #! /usr/bin/env python
22 # coding: utf-8
23 #! /usr/bin/env python
24 # coding: utf-8
25 #! /usr/bin/env python
26 # coding: utf-8
27 #! /usr/bin/env python
28 # coding: utf-8
29 #! /usr/bin/env python
30 # coding: utf-8
31 #! /usr/bin/env python
32 # coding: utf-8
33 #! /usr/bin/env python
34 # coding: utf-8
35 #! /usr/bin/env python
36 # coding: utf-8
37 #! /usr/bin/env python
38 # coding: utf-8
39 #! /usr/bin/env python
40 # coding: utf-8
41 #! /usr/bin/env python
42 # coding: utf-8
43 #! /usr/bin/env python
44 # coding: utf-8
45 #! /usr/bin/env python
46 # coding: utf-8
47 #! /usr/bin/env python
48 # coding: utf-8
49 #! /usr/bin/env python
50 # coding: utf-8
51 #! /usr/bin/env python
52 # coding: utf-8
53 #! /usr/bin/env python
54 # coding: utf-8
55
56 def openCamera(self):
57     try:
58         self.video_capture=cv2.VideoCapture(0)
59         if self.video_capture.isOpened():
60             print("Camara Encendida")
61             return True
62     except cv2.error as e:
63         print("Error iniciando la Camara: ",e)
64         return False
65
66 def publish(self):
67     mi_thread_stop.clear()
68     while not mi_thread_stop.isSet():
69         if self._topic == 'control/txv':
70             self._myCnt=self._myMsg
71             self.video_capture=cv2.VideoCapture(cam_port)
72             if self._myCnt == 'true':
73                 if self.video_capture is None or not self.video_capture.isOpened():
74                     if not self.openCamera():
75                         print("Error al iniciar la Camara")
76                         time.sleep(0.3)
77                         return
78                 self.video_capture.set(cv2.CAP_PROP_FRAME_WIDTH, 480)
79                 self.video_capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 320)
80                 result,image=self.video_capture.read()
81                 image=cv2.flip(image,1)
82                 buffer=cv2.imencode('.jpg',image,[int(cv2.IMWRITE_JPEG_QUALITY),25])
83                 b64=buffer[1]
84                 self.client.publish("IoT/Img",b64,1)
85                 time.sleep(1/2.5)
86             elif self._myCnt == 'false':
87                 self.video_capture.release()
88                 time.sleep(0.5)
89             pass
90         self.video_capture.release()
91         self.client.loop_stop()
92
93 def public_main(self,Stopic, Nmsg):
94     try:
95         self.client.publish(Stopic,Nmsg,1)
96     except AttributeError :
97         print("aun no se inicia el hilo")
98
99 def stop(self):
100     mi_thread_stop.set()

```

Anexo 7. Programación de la aplicación Android

Para la programación de la aplicación se usó el software Android Studio con el lenguaje de programación Dart.



```
1 import 'package:flutter/material.dart';
2 import 'package:smarth_tot/common/app_data.dart';
3 import 'package:smarth_tot/ui/common/faster_appbar.dart';
4
5 class MainMobile extends StatefulWidget {
6   const MainMobile({Key? key}) : super(key: key);
7   @override
8   State<MainMobile> createState() => _HomeMobile();
9 }
10 class _HomeMobile extends State<MainMobile>{
11   int _selectedIndex = 0;
12   void _onItemTapped(int index) {
13     setState() {
14       _selectedIndex = index;
15     };
16   }
17   @override
18   Widget build(BuildContext context) {
19     return Scaffold(
20       backgroundColor: Colors.black87,
21       bottomNavigationBar: BottomNavigationBar(
22         unselectedItemColor: Colors.grey,
23         selectedItemColor: Colors.teal.shade300,
24         onTap: _onItemTapped,
25         elevation: 15.0,
26         items: const [
27           BottomNavigationBarItem(
28             icon: Icon(Icons.home),
29             label: 'Inicio',
30             backgroundColor: Colors.black87), // BottomNavigationBarItem
31           BottomNavigationBarItem(
32             icon: Icon(Icons.camera),
33             label: 'Camara',
34             backgroundColor: Colors.black87), // BottomNavigationBarItem
35           BottomNavigationBarItem(
36             icon: Icon(Icons.book),
37             label: 'Registro',
38             backgroundColor: Colors.black87), // BottomNavigationBarItem
39           BottomNavigationBarItem(
40             icon: Icon(Icons.history),
41             label: 'Historial',
42             backgroundColor: Colors.black87), // BottomNavigationBarItem
43         ],
44       ),
45     );
46   }
47 }
48
49 BottomNavigationBarItem(
50   icon: Icon(Icons.settings),
51   label: 'Configuracion',
52   backgroundColor: Colors.black87), // BottomNavigationBarItem
53 ], // BottomNavigationBar
54 ), // Scaffold
55 body: SafeArea(
56   child: ConstrainedBox(
57     constraints: const BoxConstraints(maxWidth: 900),
58     child: Column(
59       children: [
60         const FasterAppBar(isMobile: true),
61         Expanded(child: screensMobile[_selectedIndex]),
62       ],),),); // Column, ConstrainedBox, SafeArea, Scaffold
```

```
1 import 'package:flutter/material.dart';
2 import 'package:smarth_tot/ui/common/historial.dart';
3
4 class HistorialMobile extends StatefulWidget {
5   const HistorialMobile({Key? key}) : super(key: key);
6   @override
7   State<HistorialMobile> createState() => _HistorialMobile();
8 }
9 class _HistorialMobile extends State<HistorialMobile>{
10   @override
11   Widget build(BuildContext context) {
12     return Stack(
13       children: [
14         Positioned(
15           bottom: 10,
16           right: 10,
17           child: FloatingActionButton(
18             onPressed: () {
19               showDialog(
20                 context: context,
21                 builder: (context) => AlertDialog(
22                   backgroundColor: const Color.fromARGB(15, 15, 15, 8.9),
23                   title: const Text("Eliminar Acciones",
24                     style: TextStyle(color: Colors.white)), // Text
25                   content: const Text("¿Estás seguro que quieres eliminar todas las acciones del historial?",
26                     style: TextStyle(color: Colors.white)), // Text
27                   actions: <Widget>[
28                     TextButton(
29                       onPressed: () => Navigator.of(context).pop(false),
30                       child: const Text("Cancelar"), // TextButton
31                     ),
32                     TextButton(
33                       onPressed: () {
34                         setState() {
35                           Historial.eliminarTodasAccion();
36                         };
37                         Navigator.of(context).pop(true);
38                       },
39                       child: const Text("Eliminar"), // TextButton
40                     ), // <Widget>[]
41                   ], // AlertDialog
42                 ),
43               );
44             },
45             child: const Icon(Icons.delete_sweep_sharp),
46           ), // FloatingActionButton, Positioned, Stack
47         ],
48       ),
49     );
50   }
51 }
52
53 child: ListTile(
54   leading: Icon(accion.icona,color: accion.color,size: 30),
55   title:Text(accion.mensaje,style: TextStyle(fontSize: 22,fontWeight: FontWeight.bold)),
56   subtitle:Text(accion.descripcion,style: TextStyle(color: Colors.black)),
57   trailing: const Icon(Icons.keyboard_arrow_right),
58   onTap: (){}), // ListTile, Container, Dismissible, Container, ListView.builder
59 ),),),); // ListTile, Container, Dismissible, Container, ListView.builder
```

```

login.dart x home_mobile.dart x camera_mobile.dart x registro_mobile.dart x
1 import 'package:flutter/material.dart';
2 import 'package:provider/provider.dart';
3 import 'package:smarth_iot/ui/common/auth_provider.dart';
4
5 class LoginScreen extends StatefulWidget {
6   @override
7   _LoginScreenState createState() => _LoginScreenState();
8 }
9 class _LoginScreenState extends State<LoginScreen> {
10   final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
11   String? _email;
12   String? _password;
13   @override
14   Widget build(BuildContext context) {
15     final authProvider = Provider.of<AuthProvider>(context);
16     return Scaffold(
17       appBar: AppBar(
18         title: Text('Login'),
19       ), // AppBar
20       body: Form(
21         key: _formKey,
22         child: Column(
23           children: <Widget>[
24             TextFormField(
25               validator: (value) {
26                 if (value!.isEmpty) {
27                   return 'Please enter your email';
28                 }
29                 return null;
30             },
31             onSave: (value) => _email = value,
32             decoration: InputDecoration(
33               labelText: 'Email',
34             ), // InputDecoration, TextFormField
35             TextFormField(
36               validator: (value) {
37                 if (value!.isEmpty) {
38                   return 'Please enter your password';
39                 }
40                 return null;
41             },
42             onSave: (value) => _password = value,
43             obscureText: true,
44             decoration: InputDecoration(
45               labelText: 'Password',
46             ), // InputDecoration, TextFormField
47             ElevatedButton(
48               child: Text('Login'),
49               onPressed: () async {
50                 if (_formKey.currentState!.validate()) {
51                   _formKey.currentState!.save();
52                   try {
53                     await authProvider.signIn(email: _email!, password: _password!);
54                     Navigator.pushReplacementNamed(context, '/home');
55                   } catch (error) {
56                     showDialog(
57                       context: context,
58                       builder: (BuildContext context) {
59                         return AlertDialog(
60                           title: Text('Login Failed'),
61                           content: Text(error.toString()),
62                           actions: <Widget>[
63                             TextButton(
64                               child: Text('OK'),
65                               onPressed: () => Navigator.pop(context),
66                             ), // TextButton, <Widget>[], AlertDialog, ElevatedButton
67                         ], // TextButton, <Widget>[], Column, Form, Scaffold
68                       ), // TextButton, <Widget>[], Column, Form, Scaffold
69                       ), // TextButton, <Widget>[], Column, Form, Scaffold
70                     Navigator.pushNamed(context, '/signup');
71                   }, // TextButton, <Widget>[], Column, Form, Scaffold

```

```

home_mobile.dart x camera_mobile.dart x registro_mobile.dart x
1 import ...
9
10 class HomeMobile extends StatefulWidget {
11   const HomeMobile({Key? key}) : super(key: key);
12   @override
13   State<HomeMobile> createState()=> _HomeMobile();
14 }
15 class _HomeMobile extends State<HomeMobile>{
16   late double _valTemp , _valHum ;
17   late TablasServices firebaseService;
18   late Map<String,FirestoreData> datosFuture;
19   List<_ChartData> chartData =[];
20   var cont=0;
21   Timer? timer;
22   ChartSeriesController? _chartSeriesController;
23   ChartSeriesController? _chartSeriesController1;
24   void addPoint(Timer timer) {
25     chartData.add(_ChartData(x: cont, y:_valHum , y2: _valTemp));
26     _chartSeriesController?.updateDataSource(addedDataIndexes: <int>[chartData.length - 1]);
27     _chartSeriesController1?.updateDataSource(addedDataIndexes: <int>[chartData.length - 1]);
28     cont+=1;
29   }
30   @override
31   void initState() {
32     super.initState();
33     cont=0;
34     _valHum=38;
35     _valTemp=38;
36     firebaseService=context.read<TablasServices>();
37     firebaseService.getData();
38     timer= Timer.periodic(const Duration(milliseconds: 500),addPoint);
39   }
40   @override
41   void dispose(){
42     timer?.cancel();
43     _chartSeriesController = null;
44     chartData.clear();
45     super.dispose();
46   }
47   @override
48   Widget build(BuildContext context) {
49     return ListView(...); // ListView
334 }
335
336 List<LineSeries<_ChartData, num>> _getDefaultLineSeries(bool sens) {
337   if (sens){
338     return
339     <LineSeries<_ChartData, num>>[...]; // <LineSeries<_ChartData, num>>[]
354   }else {
355     return
356     <LineSeries<_ChartData, num>>[...]; // <LineSeries<_ChartData, num>>[]
369   }
370 }
371
372 Widget _grafica(String titulo, bool sens,double min,double max){
373   return SfCartesianChart(...); // SfCartesianChart
409 }
410 }
411 class _ChartData {
412   _ChartData({this.x, this.y,this.y2});
413   num? x;
414   num? y;
415   num? y2;
416 }

```

```

38 class CameraMobile extends StatefulWidget {
39   const CameraMobile({key? Key}) : super(key: key);
40   @override
41   State<CameraMobile> createState() => _CameraMobile();
42 }
43
44 class _CameraMobile extends State<CameraMobile> {
45   int _counter = 0;
46   bool _stateLed = false;
47   bool _stateLed = false;
48   bool _stateLed = false;
49   final builderMqtt = MqttClientPayloadBuilder();
50   final client = mqttSetup.setup('c1933c0af5e483eb0473ef94de1a8.s2.eu.hivemq.cloud', 'Android_device123');
51   final imgBytesController = StreamController<Uint8List>();
52
53   @override
54   void initState() {
55     super.initState();
56     WidgetsBinding.instance.addPostFrameCallback((_) async {
57       await _connect(context);
58     });
59   }
60   @override
61   void dispose() {
62     super.dispose();
63     client.disconnect();
64   }
65   void _changeState(bool btn) {
66     setState(() {
67       if (btn) {
68         _stateLed = !_stateLed;
69         createNotificado(
70           "Interruptor Luz",
71           "Se cambio el estado de la luz de la camera a ${_stateLed? "ON":"OFF"}",
72           "${DateFormat('dd/MM/yyyy\\HH:mm:ss')}.format(DateTime.now())",
73           Icons.info_outline,
74           Colors.green
75         );
76       }
77     });
78   }
79
80   Widget build(BuildContext context) {
81     return Scaffold(
82       // ...
83     );
84   }
85
86   Widget buildGrid(BuildContext context) {
87     return Scaffold(
88       // ...
89     );
90   }
91
92   Future<void> _connect(context) async {
93     ProgressDialog progressDialog = ProgressDialog();
94     progressDialog.setLoadingDialog(const CircularProgressIndicator()); // CircularProgressIndicator
95     progressDialog.setMessage(const Text("Esperando conexión a su broker MQTT"));
96     progressDialog.setTitle(const Text("Conectando..."));
97     progressDialog.show();
98     client.port = 8883;
99     try {
100       await client.connect('App01', 'idapp123');
101     } on Exception catch (e) {
102       print("EXAMPLE::client exception - $e");
103       client.disconnect();
104     }
105     client.onConnected(() {
106       print("EXAMPLE::Disconnected client callback - Client connection was successful");
107     });
108     if (client.connectionStatus.state == MqttConnectionState.connected) {
109       print("EXAMPLE::Mosquitto client connected");
110       //client.subscribe('IoT/Img', MqttQos.atLeastOnce);
111       client.subscribe('control/led', MqttQos.atLeastOnce);
112       client.subscribe('control/led', MqttQos.atLeastOnce);
113       client.subscribe('control/pos', MqttQos.atLeastOnce);
114       progressDialog.dismiss();
115     } else {
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

```

1 import ...
2
3
4
5
6
7
8
9 class RegistroMobile extends StatefulWidget {
10   const RegistroMobile({key? key}) : super(key: key);
11   @override
12   State<RegistroMobile> createState() => _RegistroMobile();
13 }
14
15 class _RegistroMobile extends State<RegistroMobile> {
16   late List<DataGridRow> _dataGridRows;
17   bool opce=true;
18   String _selectedOption1 = 'Temperatura';
19   ChartSeriesController? _chartSeriesController;
20   String _selectedOption3 = 'Dia 0';
21   List<ChartData> _chartData = [];
22   final List<String> _options1 = ['Temperatura', 'Humedad'];
23   final List<String> _options2 = [
24     'Dia 0', 'Dia 1', 'Dia 2', 'Dia 3', 'Dia 4', 'Dia 5', 'Dia 6', 'Dia 7', 'Dia 8', 'Dia 9',
25     'Dia 10', 'Dia 11', 'Dia 12', 'Dia 13', 'Dia 14', 'Dia 15', 'Dia 16', 'Dia 17', 'Dia 18',
26     'Dia 19', 'Dia 20', 'Dia 21'
27 ];
28   late List<Item> items;
29   void _onOption1Changed(String? value) {
30     setState() {
31       _selectedOption1 = value ?? _options1.first;
32       if(_selectedOption1=='Temperatura'){
33         opce=true;}
34       }else{
35         opce=false;
36       }
37     });
38   void _onOption3Changed(String? value) {
39     setState() {
40       _selectedOption3 = value ?? _options2.first;
41       tablasServices.diaValue=_selectedOption3.split(' ')[1];
42     });
43   void _onSubmit() {
44     setState() {
45       if (opce){
46         chartData=List.generate(items.length,
47           (index) => _ChartData(x: items[index].fecha.toDate(), y:items[index].temp));
48       }else{
49         chartData=List.generate(items.length,
50           (index) => _ChartData(x: items[index].fecha.toDate(), y:items[index].hum)); // List.generate
51       }
52       _chartSeriesController?.updateDataSource(addedDataIndexes: <int>{chartData.length - 1});
53     });
54   }
55   late TablasServices tablasServices;
56   late Future<List<Item>> itemsFuture;
57   @override
58   void initState() {
59     super.initState();
60     tablasServices = context.read<TablasServices>();
61     tablasServices.diaValue=_selectedOption3.split(' ')[1];
62   }
63   _getDataGridRows(List<Item> items) {
64     _dataGridRows = List.generate(items.length,
65       (index) => DataGridRow(cells: [
66         DataGridCell<String>(columnName: 'date', value: DateFormat('yy-MM-dd').format(items[index].fecha.toDate())),
67         DataGridCell<String>(columnName: 'time', value: DateFormat('HH:mm:ss').format(items[index].fecha.toDate())),
68         DataGridCell<String>(columnName: 'day', value: _selectedOption3.split(' ')[1]),
69         DataGridCell<double>(columnName: 'temperature', value: items[index].temp),
70         DataGridCell<double>(columnName: 'humidity', value: items[index].hum),
71       ])); // DataGridRow, List.generate
72   }
73   @override
74   Widget build(BuildContext context) {
75     return ListView(...); // ListView
76   }
77   Widget table() {
78     return Consumer<TablasServices>(
79       builder: (context, service, child) {
80         return FutureBuilder(...); // FutureBuilder
81       }); // Consumer
82   }
83   Widget spinnerTab() {
84     return Center(...); // Center
85   }
86   Widget spinersGrar() {
87     return Center(...); // Center
88   }
89 }
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```