UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS
CARRERA DE TELECOMUNICACIONES

TRABAJO DE INTEGRACIÓN CURRICULAR

TEMA:

"IMPLEMENTACIÓN DE UN SISTEMA IOT PARA CONTROL EN EL DESARROLLO DE FORRAJE VERDE HIDROPÓNICO PARA LA CRIANZA DE CUYES EN EL CANTÓN SAN PEDRO DE HUACA"

Trabajo de titulación previo a la obtención del título en de ingeniero en telecomunicaciones

Línea de investigación:

Producción industrial y tecnología sostenible

AUTOR:

Erik Leonardo Cuasquer Cuasapaz

DIRECTOR:

Msc. Luis Edilberto Suárez Zambrano

Ibarra – Ecuador 2026

UNIVERSIDAD TÉCNICA DEL NORTE
DIRECCIÓN DE BIBLIOTECA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDADTÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
APELLIDOS Y NOMBRES:	CUASQUER CUASAPAZ ERIK LEONARDO

DATOS DE LA OBRA	
TÍTULO:	IMPLEMENTACIÓN DE UN SISTEMA IOT PARA CONTROL EN EL DESARROLLO DE FORRAJE VERDE HIDROPÓNICO PARA LA CRIANZA DE CUYES EN EL CANTÓN SAN PEDRO DE HUACA
AUTOR (ES):	CUASQUER CUASAPAZ ERIK LEONARDO
FECHA: DD/MM/AAAA	23/06/2026
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	■ **PREGRADO** □ **POSGRADO**
TITULO POR EL QUE OPTA:	INGENIERÍA EN TELECOMUNICACIONES
DIRECTOR /ASESOR:	MSC.LUIS EDILBERTO SUÁREZ ZAMBRANO

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 23 días del mes de junio de 2026.

EL AUTOR:

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
ERIK LEONARDO
CUASQUER CUASAPAZ

.....

Nombre: Erik Leonardo Cuasquer Cuasapaz

CERTIFICACIÓN DEL DIRECTOR DEL TRABAJO DEINTEGRACIÓN CURRICULAR

23 de junio de 2026

Luis Edilberto Suárez Zambrano

DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

CERTIFICA:

Haber revisado el presente informe final del trabajo de Integración Curricular, el mismo que se ajusta a las normas vigentes de la Universidad Técnica del Norte; en consecuencia, autorizo supresentación para los fines legales pertinentes.

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
LUIS EDILBERTO SUAREZ ZAMBRANO

Luis Edilberto Suárez Zambrano

C.C: 1002304291

DEDICATORIA

El trabajo logrado en el transcurso de este tiempo va dedicado, primeramente, a los motores y pilares de este logro: mis padres, por su apoyo constante e incondicional. De la misma manera, a cada uno de mis docentes, desde el primer día que pisé una escuela hasta mi director, Msc. Luis Suárez; mi asesor, Msc. Carlos Vásquez; y quien fue mi último docente, Msc. Jime Michelena.

Asimismo, dedico este trabajo al Ing. Jairo Hernández, quien me brindó su apoyo en el desarrollo agronómico del presente proyecto.

Y finalmente, este trabajo va dedicado a mi familia, quienes, cada uno a su manera, me motivaron a conseguir este logro. Espero que este no sea el último, tanto para mí como para mi familia, y que sirva de inspiración para las futuras generaciones, en especial para las personitas más pequeñas de mi familia, como mi sobrina y mis primos menores.

AGRADECIMIENTO

Aprovecho esta oportunidad para agradecer de todo corazón primeramente a Dios y a mis padres, Ana Luisa Cuasapaz y Luis Wilfrido Cuasquer Caicedo, quienes fueron mi sustento, mi apoyo incondicional y mi guía durante todo mi proceso de formación, desde mis inicios en la educación hasta hoy, que gracias a ellos tengo el honor de culminar mis estudios.

De igual manera, agradezco a mis hermanos, Anderson Cuasquer y Melany Cuasquer, mis confidentes y mi ayuda constante, quienes me animaron a seguir incluso cuando parecía que ya no tenía fuerzas; ellos siempre me impulsaron a continuar.

Quiero también expresar mi gratitud a mis docentes, quienes durante mi formación hicieron posible que hoy pueda finalizar este proceso. De forma especial, a mi tutor, el MSc. Luis Edilberto Suárez Zambrano, y a mi asesor, el MSc. Carlos Alberto Vásquez Ayala, quienes fueron mi guía en el desarrollo de este trabajo de grado. Así mismo, al Ing. Jairo Hernández, quien fue mi soporte en los temas agronómicos relacionados con el desarrollo del FVH del proyecto.

Sé que no puedo mencionar a todas las personas que aportaron su granito de arena, de manera directa o indirecta, pero si no fuera por ellas, este proceso habría sido mucho más difícil. También recuerdo con cariño a quienes ya no están, pero cuyo recuerdo me motiva a seguir adelante. En especial, a mi abuelo, quien me enseñó a estar preparado para vivir la vida, y a mi sobrino, quien me enseñó a vivir la vida sin estar preparado. Así finalizo mi agradecimiento a mis familiares y amigos que siempre estuvieron ahí.

RESUMEN

La obtención de forraje constituye uno de los factores más importantes para la sostenibilidad y rentabilidad de la crianza de cuyes en el cantón San Pedro de Huaca. Sin embargo, los métodos convencionales de producción forrajera presentan limitaciones relacionadas con la necesidad de amplias superficies de cultivo y los prolongados tiempos de cosecha. En este contexto, la presente investigación desarrolló e implementó un sistema inteligente basado en Internet de las Cosas (IoT) e inteligencia artificial para automatizar el monitoreo y control de las condiciones ambientales requeridas en la producción de Forraje Verde Hidropónico (FVH) de trigo en espacios pequeños.

La solución integra sensores de temperatura, humedad relativa, luminosidad, humedad del sustrato y pH, conectados a un nodo sensor basado en ESP32, hacia una Raspberry Pi 5 configurada como Edge Gateway. La plataforma incorpora servicios de almacenamiento local, visualización en tiempo real mediante Node-RED, acceso remoto desde una aplicación móvil desarrollada en Flutter y notificaciones mediante Telegram. Paralelamente, se implementó un modelo de aprendizaje supervisado basado en Árboles de Decisión para automatizar el proceso de riego. Permitiendo mantener condiciones adecuadas para el desarrollo del cultivo y optimizar el uso de los recursos hídricos.

Palabras clave: Internet de las Cosas (IoT); Forraje Verde Hidropónico; Agricultura Inteligente; Árboles de Decisión; Automatización Agrícola; ESP32; Raspberry Pi 5; Crianza de Cuyes.

ABSTRACT

The obtaining of forage constitutes one of the most important factors for the sustainability and profitability of guinea pig rearing in the San Pedro de Huaca canton. However, conventional forage production methods present limitations related to the need for large cultivation areas and prolonged harvest times. In this context, the present research developed and implemented an intelligent system based on the Internet of Things (IoT) and artificial intelligence to automate the monitoring and control of the environmental conditions required in the production of wheat hydroponic green forage (HGF) in small spaces.

The solution integrates temperature, relative humidity, luminosity, substrate moisture, and pH sensors, connected to an ESP32-based sensor node, towards a Raspberry Pi 5 configured as an Edge Gateway. The platform incorporates local storage services, real-time visualization through Node-RED, remote access from a mobile application developed in Flutter, and notifications through Telegram. In parallel, a supervised learning model based on Decision Trees was implemented to automate the irrigation process. Allowing to maintain adequate conditions for crop development and optimize the use of water resources.

Keywords: Internet of Things (IoT); Hydroponic Green Forage; Smart Agriculture; Decision Trees; Agricultural Automation; ESP32; Raspberry Pi 5; Guinea Pig Rearing.

ÍNDICE

CAPÍTULO I: Antecedentes 17

1.1 Tema 17

1.2 Problema 17

1.3 Objetivos 23

1.3.1 Objetivo General 23

1.3.2 Objetivos Específicos 23

1.4 Alcance 24

1.4.1 Fase uno: Requisitos 25

1.4.2 Fase dos: Diseño del sistema 26

1.4.3 Fase tres: Implementación 27

1.4.4 Fase cuatro: Pruebas 28

1.4.5 Fase cinco: Mantenimiento 28

1.5 Justificación 28

CAPÍTULO II: Fundamentación Teórica 30

2.1 Introducción 30

2.2 Hidroponía 30

2.2.1 Historia de la Hidroponía 31

2.2.2 Forrajes Verdes Hidrópicos..... 32

2.2.3 Importancia de la Agricultura Hidropónica en la crianza de cuyes. 33

2.2.4 Tipos de forrajes para Hidroponía..... 34

2.2.5 Estudios comparativos de métodos tradicionales y modernos en la Producción de Forrajes..... 38

2.3 El Cuy..... 39

2.4 Sistema IoT..... 40

2.4.1 Composición de un Sistema IoT 40

2.4.2 Funcionamiento básico de un Sistema IoT 41

2.4.3 Dispositivos en sistemas IoT..... 43

2.5 Beneficios del uso de tecnologías IoT en la agricultura. 45

2.6 Algoritmos de inteligencia artificial aplicados a la agricultura. 45

2.7 Sistemas de comunicación y protocolos en IoT. 48

2.8 Plataformas de Software para Monitoreo..... 50

2.9 Bases de Datos..... 52

CAPÍTULO III: Desarrollo de la propuesta 54

3.1 Introducción..... 54

3.2 Descripción de la metodología 54

3.3 Requerimientos..... 55

3.3.1 Análisis situacional 56

3.3.2 Análisis requerimientos 58

3.3.3 Requerimientos operacionales y de usuario 61

3.3.4 Requerimientos del sistema 63

3.3.5 Requerimientos de la arquitectura 66

3.4 Diseño del sistema 69

3.4.1 Selección de hardware 69

3.4.2 Selección de software 94

3.4.3 Estructuración del sistema 101

3.5 Calculo de consumo de corriente del sistema 110

3.6 Lógica del sistema 113

3.6.1 Estructuración del sistema IoT 113

3.7 Configuraciones del sistema 125

3.7.1 Configuración sensores 125

3.7.2 Configuración nodo sensor 133

3.7.3 Configuración de Edge Gateway 135

3.7.3.1 Programación 135

3.7.4 Árbol de decisión 146

3.7.5 Plataforma monitoreo 150

3.8 Aplicación móvil (FVH control app) 153

3.9 VPN 155

CAPÍTULO IV 157

4.1 Implementación 158

4.1.1 Construcción del invernadero para FVH. 158

4.1.2 Adquisición de la bomba de agua para el riego 159

4.1.3 Implementación de los aspersores en las estructuras. 160

4.1.4 Implementación de componentes eléctricos 164

4.1.5 Implementación se componentes electrónicos 168

4.1.6 Comprobación de sensado del sistema implementado 172

4.1.7 Análisis del proceso de comunicación del sistema 178

4.1.8 Diagrama de comunicación final del sistema 193

4.1.9 Comprobación de monitoreo del sistema 195

4.2 Pruebas 197

4.2.1 Planificación del cultivo FVH 197

4.2.2 Primera etapa de pruebas 201

4.3 Pruebas de Hardware/Software 208

4.3.1 Pruebas de Acoplamiento de los Componentes 211

4.3.2 Pruebas de Comunicación de componentes del sistema 212

4.3.3 Pruebas de Funcionamiento del Nodo Central 213

4.3.4 Pruebas de Funcionamiento de Conexión Remota con Dispositivo Móvil..... 215

4.4 Medición de Parámetros del Estado Nutricional Hidropónico..... 220

4.5 Evaluación de resultados..... 222

4.5.1 Análisis del desarrollo del FVH de Trigo..... 222

4.5.2 Altura de la Planta..... 223

4.5.3 Coloración de Hoja del FVH de Trigo..... 227

4.5.4 Frondosidad del FVH de Trigo..... 229

4.5.5 Desarrollo de Raíz..... 231

4.5.6 Análisis del Consumo de Agua en el Sistema Hidropónico..... 233

4.6 Comparativa del Cultivo de FVH IoT con método tradicional..... 240

4.7 Mantenimiento..... 246

4.7.1 Recalibración de Sensores..... 247

4.7.2 Actualización de métodos de comunicación..... 248

4.7.3 Mantenimiento de materiales físicos de producción de FVH..... 249

4.8 Costos del sistema..... 250

4.8.1 Costo de Hardware..... 250

4.8.2 Costo de Infraestructura..... 251

4.8.3 Costos de Ingeniería..... 252

4.8.4 Costos de Software..... 253

4.8.5 Costo Global del Sistema..... 254

4.9 Beneficios del sistema 255

CONCLUSIONES 257

RECOMENDACIONES 258

BIBLIOGRAFÍA 259

ÍNDICE DE FIGURAS

Figura 1 *Modelo Cascada* 24

Figura 2 *Arquitectura del sistema* 27

Figura 3 *Ubicación de la vivienda* 56

Figura 4 *Esquemático del área de implementación* 58

Figura 5 *Placa ESP-WROOM-32D seleccionada para nodo sensor y actuador* 72

Figura 6 *Placa seleccionada para Edge Gateway* 76

Figura 7 *Sensor de temperatura seleccionado es el* Modelo *DTH 22* 79

Figura 8 *Sensor Capacitivo v2.0* 82

Figura 9 *Sensor seleccionado para monitorear luminosidad* 85

Figura 10 *Selección del Sensor para pH de agua: Modelo PH-4502C* 88

Figura 11 *Bomba de riego modelo Paolo PC-122* 91

Figura 12 *Contactor seleccionado para el sistema.* 94

Figura 13 *Sistema Operativo para la placa Rapsberri pi 5 B* 96

Figura 14 *Logo de la base de datos SQLite* 98

Figura 15 *Plataforma de monitoreo "Node-Red"* 100

Figura 16 *Logotipo de app Telegram* 101

Figura 17 *Estantes metálicos para FVH* 102

Figura 18 *Microaspersores que serán utilizados para el riego* 103

Figura 19 *Diagrama de Bloques del sistema* 104

Figura 20 *Arquitectura general del sistema de FVH* 109

Figura 21 *Diagrama de flujo del sensado de humedad* 114

Figura 22 *Diagrama de flujo del sensado de temperatura* 115

Figura 23 *Diagrama de flujo del sensado de luz* 116

Figura 24 *Diagrama de flujo de lectura del sensor de pH* 118

Figura 25 *Diagrama de flujo del nodo sensor* 120

Figura 26 *Diagrama de control del riego* 121

Figura 27 *Diagrama de Flujo de la placa central* 124

Figura 28 *Conexiones del sensor de humedad* 126

Figura 29 *Código para el sensor de Humedad* 127

Figura 30 *Conexiones del sensor DTH22 con la placa ESP 32D* 128

Figura 31 *Código para el sensor DTH22* 128

Figura 32 *Conexiones para el sensor BH1750 con la placa ESP 32D* 129

Figura 33 *Código para el Sensor de luz* 130

Figura 34 *Conexiones del sensor de pH* 131

Figura 35 *Código implementado para sensor de pH* 132

Figura 36 *Conexión final de los sensores a la placa ESP 32D* 133

Figura 37 *Código para el nodo sensor y actuador* 134

Figura 38 *Archivos de configuración de la placa central* 135

Figura 39 *Archivo de subscriptor de servicio MQTT* 136

Figura 40 *Librerías necesarias para el árbol de decisión e Inicialización* 137

Figura 41 *Carga datos para entrenamiento* 139

Figura 42 *Reglas para el entrenamiento del árbol.* 140

Figura 43 *Entrenamiento del árbol de decisión.* 142

Figura 44 *Ejemplos de predicción* 143

Figura 45 *Archivo del sistema de inferencia* 144

Figura 46 *Código para el control mediante mensajes en MQTT* 145

Figura 47 *Visualización del Árbol de decisión entrenado* 147

Figura 48 *Gráfico de dispersión respecto a decisiones.* 148

Figura 49 *Rutas de decisión* 149

Figura 50 *Nodos de configuración de la plataforma de monitoreo* 150

Figura 51 *Extracción de datos de la base de datos* 151

Figura 52 *Dashboard final en la plataforma de node-red* 152

Figura 53 *Interfaz App móvil* 153

Figura 54 *Series temporales de los datos de los sensores* 154

Figura 55 *Historial de registros en app móvil* 155

Figura 56 *Tailscale estructura* 156

Figura 57 *Construcción del Invernadero para desarrollo de FVH* 159

Figura 58 *Bomba de riego Paolo* 160

Figura 59 *Implementación de microaspersores* 161

Figura 60 *Distribución de mangueras para microaspersión* 161

Figura 61 *Conexión de mangueras a la bomba* 163

Figura 62 *Tanque de alimentación de sistema de riego* 163

Figura 63 *Protección del sistema* 165

Figura 64 *Implementación del Contactor* 166

Figura 65 *Conexión de la Bomba de riego* 167

Figura 66 *Alimentación para el Nodo y Sensores* 168

Figura 67 *Caja del Nodo de sensado y actuación* 169

Figura 68 *Sensores y conexiones externas de la caja del Nodo* 170

Figura 69 *Ubicación del sensor de pH* 171

Figura 70 *Ubicación de los sensores de humedad* 172

Figura 71 *Obtención de datos en los sensores de humedad* 174

Figura 72 *Funcionamiento del Sensor de Temperatura* 175

Figura 73 *Funcionamiento del Sensor de Luminosidad BH1750* 176

Figura 74 *Medición de pH con cinta medidora* 177

Figura 75 *Funcionamiento del Sensor en pH 4* 178

Figura 76 *Funcionamiento del Sensor en pH 7.5* 178

Figura 77 *Interfaces de comunicación del Edge Gateway* 181

Figura 78 *Conexión de la placa ESP32-D a la red Wi-Fi* 181

Figura 79 *Flujo de conexión y suscripción MQTT* 182

Figura 80 *JSON elaborado por el nodo para envío al Gateway* 183

Figura 81 *Comunicación mediante MQTT* 184

Figura 82 *Guardado de datos en SQLite* 186

Figura 83 *Control de activación o desactivación de riego* 187

Figura 84 *Diagrama de flujo de comunicación en la capa de aplicación.* 190

Figura 85 Diagrama de Comunicación 194

Figura 86 *Envío de alertas de Telegram* 195

Figura 87 *Dasboard funcional* 196

Figura 88 *Preparación para germinación en agua con cal* 202

Figura 89 *Selección de semillas aptas para proceso* 203

Figura 90 *Oreado de las semillas* 203

Figura 91 *Ubicación Bandejas en estantes* 204

Figura 92 *Bandejas al tercer día de sembradas* 205

Figura 93 *Medición de brotes de trigo* 206

Figura 94 *Crecimiento de primeros días del fvh* 206

Figura 95 *FVH listo para la cosecha.* 207

Figura 96 *Consumo del producto final de fvh.* 208

Figura 97 *Estadísticas de registros filtrados por fechas* 217

Figura 98 *pH monitoreado por un periodo de tiempo* 217

Figura 99 *Humedad monitoreada por un periodo de tiempo* 219

Figura 100 *Temperatura monitoreada por un periodo de tiempo* 220

Figura 101 *Altura primeros días del cultivo tradicional* 223

Figura 102 *Plántulas desarrolladas en segunda semana del método tradicional* 224

Figura 103 *Medición de plantas en la tercera semana* 225

Figura 104 *Frondosidad con el método tradicional.* 229

Figura 105 *Frondosidad de FVH* 230

Figura 106 *Desarrollo de la raíz método tradicional* 231

Figura 107 *Desarrollo de raíces del FVH* 232

Figura 108 *Fluctuación del consumo semanal de agua* 236

Figura 109 *Acumulación consumo general de agua* 236

Figura 110 *Tiempo de crecimiento de los cultivos* 243

Figura 111 *Comparativa eficiencia del agua* 244

Figura 112 *Comparativa de frecuencia de cosechas* 245

Figura 113 *Uso del Suelo* 246

ÍNDICE DE TABLAS

Tabla 1 *Comparación entre Agricultura Hidropónica y Agricultura Tradicional* 20

Tabla 2 *Aporte nutricional del forraje verde hidropónico de avena.* 35

Tabla 3 *Constituyentes porcentuales de las diferentes partes de la cebada.* 36

Tabla 4 *Valores nutricionales del trigo por 100g* 37

Tabla 5 *Listado de Stakeholders del proyecto* 59

Tabla 6 *Nomenclatura utilizada para requerimientos* 60

Tabla 7 *Priorización de Requerimientos mediante niveles* 60

Tabla 8 *Requerimientos operacionales y de usuario* 61

Tabla 9 *Requerimientos del sistema IoT* 63

Tabla 10 *Requerimientos de la arquitectura del sistema IoT* 66

Tabla 11 *Características de las placas para nodo sensor y actuador* 70

Tabla 12 *Selección de nodo sensor y actuador* 71

Tabla 13 *Características de placas para Edge Gateway* 73

Tabla 14 *Selección de placa que funcionará como Edge Gateway* 75

Tabla 15 *Características de posibles sensores de temperatura* 77

Tabla 16 *Requisitos para sensor de Temperatura* 78

Tabla 17 *Características de las opciones de sensor de humedad* 80

Tabla 18 *Requisitos para selección del sensor de humedad* 81

Tabla 19 *Características de las opciones para sensor de luminosidad* 83

Tabla 20 *Requisitos para selección del sensor de luz solar* 84

Tabla 21 *Características sensores de pH* 86

Tabla 22 *Requisitos para selección del sensor de ph* 87

Tabla 23 *Características de bomba para riego* 89

Tabla 24 *Requisitos para selección de Bomba de riego* 90

Tabla 25 *Características de modelos de bomba* 92

Tabla 26 *Evaluación de requerimientos para contactor* 93

Tabla 27 *Evaluación de software para nodo sensor* 95

Tabla 28 *Evaluación de distintas bases de datos* 97

Tabla 29 *Evaluación para la plataforma de monitoreo* 99

Tabla 30 *Consumo de corriente en DC de los componentes del sistema.* 110

Tabla 31 *Consumo de actuadores en AC* 111

Tabla 32 *Comunicación de sensores hacia al esp 32* 180

Tabla 33 *Comunicación Nodo y Gateway* 184

Tabla 34 *Capa apoyo Comunicación* 188

Tabla 35 *Comunicación capa aplicación* 192

Tabla 36 *Planificación de cultivo inicial* 199

Tabla 37 *Planificación constante del sistema* 200

Tabla 38 *Calculo cosecha esperada del sistema* 201

Tabla 39 *Cronograma de pruebas del funcionamiento del sistema* 209

Tabla 40 *Pruebas de acoplamiento y funcionalidad partes del nodo sensor.* 211

Tabla 41 *Pruebas de comunicación entre nodos* 213

Tabla 42 *Pruebas de nodo Edge Gateway* 214

Tabla 43 *Pruebas de comunicación móvil* 215

Tabla 44 *Rangos de Variables Ambientales y Nutritivas Medidos por el Sistema IoT-FVH* 221

Tabla 45 *Tabla de crecimiento del método tradicional* 225

Tabla 46 *Crecimiento periódico del FVH de Trigo* 226

Tabla 47 *Referencia de pigmentación de la planta* 228

Tabla 48 *Consumo de agua* 234

Tabla 49 *Tabla Comparativa de métodos de Cultivo* 241

Tabla 50 *Costos de Hardware* 250

Tabla 51 *Costos de Infraestructura* 251

Tabla 52 *Costos de Ingeniería* 252

Tabla 53 *Costos de Software* 253

Tabla 54 *Costos Globales* 254

CAPÍTULO I: Antecedentes

1.1 Tema

IMPLEMENTACIÓN DE UN SISTEMA IOT PARA CONTROL EN ELDESARROLLO DE FORRAJE VERDE HIDROPÓNICO PARA LA CRIANZA DE CUYES EN EL CANTÓN SAN PEDRO DE HUACA.

1.2 Problema

La mayor demanda de cuyes se concentra principalmente en las provincias interandinas dela sierra ecuatoriana, como Tungurahua, Azuay, Cotopaxi, Pichincha, Chimborazo e Imbabura. Según el último censo agropecuario, la población de cuyes fue de 5’067.049 animales, de los cuales el 97 corresponde a crianza familiar y tradicional, y el restante a exportaciones tecnificadas (Reyes et al., 2021). Esto evidencia que Carchi no es un competidor significativo en la producción de cuyes. Sin embargo, aunque la demanda de este producto es mayor en las provincias mencionadas, la finalidad no es necesariamente la comercialización, como señalan (Camacho et al., 2022). A nivel nacional, los productores destinan el 35 % de los cuyes para autoconsumo, el 6 % para comercialización, y el 59 % realiza esta actividad de manera mixta, tanto para autoconsumo como para comercio. Aunque la producción de cuyes en Carchi no es tan amplia comercialmente, las personas que se dedican a esta actividad en el cantón San Pedro de Huaca enfrentan diversas dificultades para obtener el forraje necesario para sus cuyales.

Una de las primeras dificultades que enfrentan los pequeños productores de cuyes es que los habitantes del cantón San Pedro de Huaca están más interesados en la ganadería y la agricultura enfocada en la producción de papa. Según datos del Censo de Población y Vivienda, el 54,77 % de la población se dedica a la agricultura, ganadería, silvicultura y pesca,

mientras que el 7 % de la PEA (Población Económicamente Activa) se dedica al comercio al por mayory menor, el 3,10 % trabaja en actividades de construcción, el 4,39 % en transporte, el 3,81% en la industria manufacturera y el 11,21 % realiza actividades no declaradas (Velarde, 2013).

Lo mencionado anteriormente evidencia la falta de espacios dedicados a la producción de forrajes que cumplan los requerimientos nutricionales adecuados para la crianza de cuyes, debido a que los terrenos del sector se destinan a otros fines. El uso del suelo en esta zona se destina principalmente a actividades de agricultura y ganadería, ya que el 64,18 % del territorio se utiliza en la producción de cultivos y pasto, el 17,50 % está cubierto por bosques y el 17,30 % es páramo (García et al., 2018).

De los 70,9 km² que componen la extensión del territorio del cantón San Pedro de Huaca, según su plan de desarrollo y ordenamiento territorial, aproximadamente 45,5 km² están destinados exclusivamente a la producción agraria y ganadera, dejando sin espacios a los pequeños productores que se dedican a criar cuyes. El sector estudiado se caracteriza por tener un clima frío, con precipitaciones lluviosas y secas distribuidas a lo largo del año. Los meses de abril ynoviembre son los más fríos, con temperaturas que no superan los 5 °C y lluvias durante todoel mes páramo (García et al., 2018).

Esta zona, al estar ubicada geográficamente en el callejón interandino que comprende los valles andinos y los páramos, incluye 10 provincias distribuidas de norte a sur. Su clima varía entre los 8°C y los 20°C, con una gradiente de temperatura de -5°C por cada 100 m de altura (Varela & Ron, 2022). Debido a estos cambios climáticos, resulta complicado mantener una plantación de hierba eficiente, ya que este cultivo necesita temperaturas y condiciones específicas para un desarrollo óptimo.

Esta situación presenta otra dificultad para producir forrajes adecuados para una buena crianza de cuyes, dado que estos animales dependen principalmente de su alimentación, la cual representa entre el 70 % y el 80 % del costo de producción. Los forrajes verdes son una parte crucial de la dieta de los cuyes, proporcionando las vitaminas, minerales y fibra necesariospara su correcto crecimiento y desarrollo (Montes, 2012).

Los cobayos poseen una digestión fermentativa postgástrica, lo que implica que necesitan un aporte constante de fibra en sus raciones alimenticias. Esta fibra debe constituir aproximadamente el 15 % de la fibra cruda (FC) para favorecer la digestibilidad de otros nutrientes, ya que retrasa el paso del contenido alimenticio a través del tracto digestivo (Farinango & Milton, 2023). Si se les proporciona una alimentación insuficiente en calidad y cantidad, puedenpresentarse diversos trastornos. Para asegurar que los cuyes tengan una buena producción y crezcan rápidamente, se les debe suministrar un alimento adecuado a sus requerimientos nutritivos (Padilla, 2021).

Es por eso que la implementación de sistemas de FVH (Forraje Verde Hidropónico) en el sector resulta una solución efectiva a estas dificultades y demandas. Proveer forrajes adecuados a la producción de cuyes mediante este sistema presenta varias ventajas, incluyendo su alto valor nutritivo y digestibilidad. Un kilo de semilla se transforma en más de seis kilos de bio- masa vegetal. Además, tiene un corto tiempo de producción, completándose en 12 a 15 días, permitiendo dos cosechas al mes. El proceso también ahorra significativamente agua y reducecostos operativos, ya que requiere menos terreno y recursos en comparación con el cultivo tradicional. El forraje verde hidropónico es limpio e inocuo, mejorando la condición corporal, la producción de carne, la fertilidad y la salud del ganado,

además de aumentar las ganancias en conejos y cuyes al sustituir el alimento balanceado (García, 2013).

De acuerdo a la **Tabla 1** se presenta una comparativa de las características y diferencias de la producción de FVH con el método tradicional de acuerdo a (García, 2013).

Tabla 1

Comparación entre Agricultura Hidropónica y Agricultura Tradicional

FACTOR	AGRICULTURA HIDROPÓNICA	AGRICULTURA TRADICIONAL
Tierra	No se requiere	Sí se requiere
Maquinaria y Equipo	Automatizado	Exigente y costosa
Semillas	Mínimo desperdicio	Mayor desperdicio
Mano de obra	Muy poca	Bastante
Calidad Producto	Excelente	Variable
Número de Plantas	Limitado únicamente por la iluminación	Limitado por la nutrición que el suelo puede proporcionar a las plantas y por la disponibilidad de luz.
Control de malas hierbas	No existe	Sí existe

Enfermedades y parásitos del suelo	No hay presencia de enfermedades, insectos o animales en el medio de cultivo, tampoco existe enfermedades en las raíces, no requiere rotación de cultivos.	Gran presencia de enfermedades, insectos, nematodos en el suelo, es necesario una rotación de cultivos.
Estado sanitario	Al no añadir agentes biológicos a las plantas no existen agentes patógenos en ellas.	Los restos orgánicos que se utilizan como fertilizantes son fuente del inóculo de plagas y enfermedades.

Nota: *La tabla presenta una comparación entre las técnicas de producción de la agricultura tradicional y conhidroponía.*

Por lo tanto, la necesidad de disponer de un espacio para producir alimentos de manera constante y controlada, además de estar libre de enfermedades y otras inclemencias, puede ser satisfecha con la implementación de un sistema IoT. Este sistema aseguraría las condiciones ideales para producir forraje hidropónico, aprovechando los beneficios nutricionales de los FVH (Forrajes Verdes Hidropónicos) y el poco espacio que estos requieren para su producción, como lo afirman (Ku et al., 2016). Al no estar en contacto con la tierra, estos cultivos estarían libres de bacterias, parásitos, contaminación y hongos.

Además, permite el aprovechamiento del entorno, permitiendo organizar los cultivos de diversas formas, incluyendo la opción de una organización vertical. No se necesita maquinaria agrícola, lo que los hace más manejables e incluso permite la automatización. De esta manera, se brinda una mayor competitividad a los pequeños productores de cuyes del cantón San Pedro de Huaca.

1.3 Objetivos

1.3.1 Objetivo General

Implementar un sistema IoT que controle el desarrollo de forraje verde hidropónico (FVH), de manera automática, para apoyar a los pequeños productores en la crianza de cuyes en el cantón San Pedro de Huaca.

1.3.2 Objetivos Específicos

- Laborar un estado del arte sobre la producción de Forraje Verde Hidropónico (FVH) enel cantón San Pedro de Huaca, evaluando sus posibles beneficios en la crianza de cuyesy en el sector ganadero.
- Establecer los requerimientos para desarrollar un sistema IoT, considerando factores como temperatura, humedad, pH del agua y otros aspectos necesarios para el desarrollode FVH.
- Implementar el sistema IoT capaz de llevar el control automático del ambiente, para que el FVH pueda desarrollarse adecuadamente.
- Realizar pruebas de control y correcto funcionamiento del sistema ya implementado,con el fin de obtener resultados de su funcionamiento.

1.4 Alcance

En el presente proyecto de titulación, se busca implementar un sistema IoT con algoritmos de inteligencia artificial. Según (Rouhiainen, 2018), este tipo de algoritmos permiten que los ordenadores o las máquinas tengan la capacidad de aprender sin estar específicamente programados para ello. Un ejemplo típico serían las sugerencias o predicciones en situaciones particulares. Garantizando la producción de FVH de manera consistente y saludable para la crianzade cuyes.

El modelo en el cual se va a trabajar para el desarrollo del proyecto de grado será enbase al modelo de cascada, como podemos observar en la **Figura 1**, buscando definir los requisitos necesarios para implementar el sistema, realizar el diseño más adecuado de acuerdo con el entorno, realizar las pruebas de funcionamiento correspondientes, proceder con la implementación del sistema y, finalmente, analizar y dar mantenimiento adecuado para garantizar lavida útil del sistema.

Figura 1

Modelo Cascada



Nota: *Este ejemplo se basa en el método de cascada según (Laoyan, 2024).*

1.4.1 Fase uno: Requisitos

Se llevará a cabo una minuciosa investigación bibliográfica sobre los requerimientos ambientales para el desarrollo de Forrajes Verdes Hidropónicos (FVH) y se identificarán las variables ambientales que deben analizarse para garantizar un ambiente adecuado para este tipode cultivos. Se investigará la situación actual de los pequeños productores de cuyes y cómo obtienen el sustento necesario para la cría de sus animales, mediante la recopilación de información en instituciones del sector, como municipios e investigaciones previas, en repositoriosde universidades del Ecuador e instituciones internacionales.

Se realizará una comparativa de los dispositivos que mejor se adapten a estas necesidades, investigando los precios y características de cada uno de estos componentes para establecer los dispositivos adecuados a emplear posteriormente. Se plantea estudiar la mejor opción de semillas para la producción del FVH en el que el trigo podría ser una excelente debido a su notable respuesta a las variaciones en la densidad de siembra y su capacidad para producir un rendimiento significativo.

Ya que en los experimentos hechos por Felipe Sánchez Del Castillo los resultados de su experimento 1, se observó que, con una alta densidad de siembra, el trigo alcanzó una mayor altura de la planta debido a la elongación celular provocada por la menor incidencia de radiación por planta (F. S. Del Castillo et al., 2013). Esto es beneficioso para la producción de FVH, ya que se puedeobtener una mayor cantidad de forraje en un espacio reducido.

1.4.2 Fase dos: Diseño del sistema

El sistema estará compuesto por microcontroladores y placas de administración, conecta-dos a una red de sensores que tomarán los valores de las variables previamente identificadas como importantes en el desarrollo del sistema IoT. De esta manera, se mantendrá un constante monitoreo del ambiente. Todo esto se logrará mediante la implementación de algoritmos de aprendizaje de inteligencia artificial, garantizando que el sistema sea lo más preciso posible ypueda evitar posibles afectaciones al FVH.

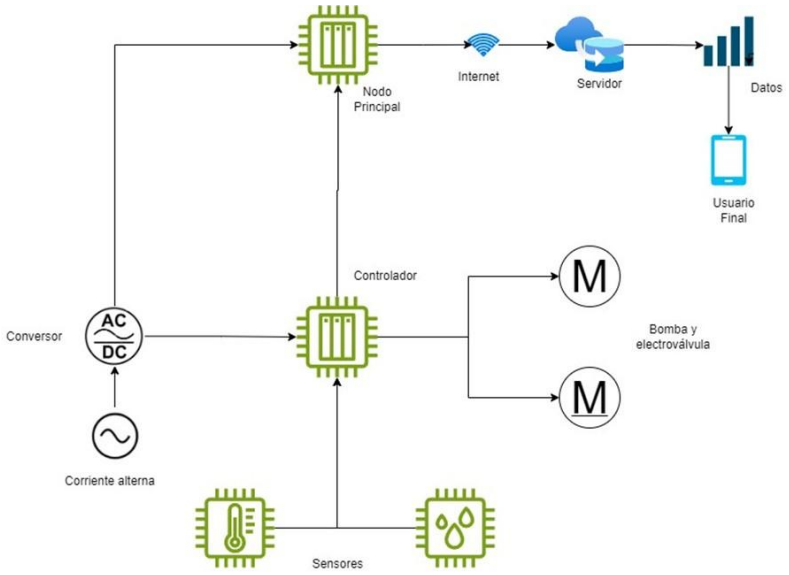
Se aprovechará el espacio disponible dentro de un área reducida para producir la misma cantidad de alimento que en un área amplia de terreno, proponiendo de 3 a 4 niveles en los cuales se producirán forrajes verdes hidropónicos. La hidroponía es una modalidad de manejo de plantas que permite su cultivo sin suelo. Con esta metodología se pueden producir principalmente plantas herbáceas, aprovechando además áreas o sitios no convencionales típicos deambientes urbanos, como las grandes ciudades (Albuja et al., 2021).

El sistema estará en constante comunicación con el usuario final, quien podrá monitorear el estado de los cultivos de FVH en tiempo real mediante una plataforma de base de datos donde se presentará el estado del sistema.

En la **Figura 2** se muestra la arquitectura con la que se pretende comprender la estructuración final del sistema con el fin de que este pueda funcionar de una manera óptima en las distintas situaciones.

Figura 2

Arquitectura del sistema



Nota: *Se realizó el diseño en un diagramador en línea llamado Draw.io.*

1.4.3 Fase tres: Implementación

Se buscará un lugar adecuado para implementar los estantes y adaptar un espacio para ubicar las placas ya programadas correctamente, conectadas con los respectivos sensores y actuadores, de acuerdo con el diseño del sistema. La ubicación se determinará teniendo en cuenta la sectorización de cada punto designado para el riego o la colocación de los aspersores de un kit de nebulización, que se activará una vez que se enciendan los actuadores (electroválvulas).Todo esto se realizará siguiendo las normas de seguridad para garantizar la correcta implementación del sistema y el bienestar del instalador.

1.4.4 Fase cuatro: Pruebas

Se realizarán pruebas de estabilidad del sistema para verificar que cada sensor y actuador funcionen correctamente de acuerdo con el entorno en el que se implementen. Además, se evaluarán los aspectos del tipo de germinado que se utilizará para la producción de FVH, manteniendo un análisis del estado del sistema. Esto permitirá evaluar y obtener las conclusiones necesarias para actuar en caso de presentarse inconvenientes.

1.4.5 Fase cinco: Mantenimiento

Finalmente, se desarrollará un manual de usuario final, estableciendo métodos de guía comprensible e interactiva, ya que los usuarios no poseen conocimientos fundamentales en esta área. El objetivo es que el sistema pueda ser usado correctamente y se le dé el mantenimiento adecuado para prolongar su vida útil.

1.5 Justificación

El presente trabajo de grado se justifica al alinearse con el Objetivo de Desarrollo Sostenible (ODS) número 9, de acuerdo con (Guterres, 2023). El desarrollo social y la acción por el clima depende en gran medida de las inversiones en infraestructuras, el desarrollo industrial sostenible y el progreso tecnológico. Ante la rápida evolución del panorama económico mundial y el aumento de las desigualdades, el crecimiento sostenido debe implicar una industrialización que, en primer lugar, haga accesibles las oportunidades a todas las personas y, ensegundo lugar, se apoye en la innovación y en infraestructuras resistentes.

En el contexto del Carchi, los procesos de producción de forrajes tienden a ser ineficientes y, a menudo, rústicos. Por lo tanto, se busca modernizar estos procesos mediante la implementación de una infraestructura tecnológica. Específicamente, se propone integrar el

Internet de las Cosas (IoT) en la producción de FVH para la cría de cuyes. Este sistema permitirá monitorear y controlar de manera las condiciones ambientales para el cultivo de forraje hidropónico,mejorando así la productividad y la calidad del alimento para los cuyes, optimizando el uso derecursos como el agua y la energía.

El impacto económico del proyecto se deriva de la mejora en la productividad de FVH para los pequeños criadores de cuyes. La producción de alimento en espacios limitados permitirá a estos criadores ganar competitividad en la venta de los animales, generando ingresos adicionales para sus familias. Al vender un producto cárnico rico en proteínas, saludable y de buenacalidad, como lo evidencia el estudio realizado en el sector Tambo, el análisis de Beneficio - Costo demuestra que es mayor que uno. Específicamente, el valor de 1,60 indica que, por cadadólar invertido, se recupera dicho valor y se obtiene una ganancia adicional de $0.60 (Chima & Luna, 2022).

Además, el proyecto influirá en el sector agrícola tradicional, fomentando la adopción de prácticas sostenibles y modernas mediante el uso de IoT, mejorando así la eficiencia y sostenibilidad en la región.

Además, el proyecto puede tener un impacto en la inclusión del conocimiento tecnológico y digital, alineado con el ODS 9.c. Beneficiará tanto a pequeños como a grandes agricultores del sector, proporcionándoles información sobre el funcionamiento del sistema IoT implementado, lo que podría despertar su interés en aplicar estas tecnologías en sus métodos de producción.

CAPÍTULO II: Fundamentación Teórica

2.1 Introducción

El presente capítulo aborda los fundamentos que sustentan la viabilidad y pertinencia de implementar el proyecto en el entorno específico del beneficiario. Esta base teórica se construye mediante una revisión bibliográfica exhaustiva de literatura tecnológica, agronómica y agropecuaria.

Se examinan temáticas clave como el diseño y gestión de bases de datos, los algoritmos de aprendizaje automático y predicción, así como la arquitectura y aplicación de sistemas IoT en contextos de producción vegetal. Este análisis fundamenta la selección técnica de los equipos y componentes necesarios para la aplicabilidad del proyecto.

Paralelamente, se estudian los métodos de crianza de cuyes (Cavia porcellus), sus características zootécnicas, valor nutricional y requerimientos específicos. De igual forma, se analiza el valor nutricional de los insumos para la producción de Forraje Verde Hidropónico (FVH).

En conjunto, la revisión establece un marco de referencia sólido y claro que sirve de fundamento para el desarrollo subsiguiente del sistema propuesto.

2.2 Hidroponía

La hidroponía es una técnica de cultivo que optimiza el uso de recursos, incrementa la producción de alimentos y fomenta la sostenibilidad agrícola, particularmente en regiones áridascon escasez de agua y tierra cultivable limitada. Este método permite cultivar plantas en ambientes controlados sin necesidad de suelo, facilitando la producción continua durante todo elaño, especialmente en áreas urbanas y áridas. Estudios han mostrado que la hidroponía

per- mite un crecimiento vegetal significativamente más rápido, con incrementos que van del 30 al 500 por ciento respecto a la agricultura convencional, resultando en mayor productividad ymenor uso de fertilizantes (Zhu et al., 2023).

2.2.1 Historia de la Hidroponía

A finales del siglo XVII, durante la última década, un hombre se acercó a las aguas de Hyde Park en Londres y recogió un poco en un recipiente. Probablemente, nadie notó su acción. Ese día en 1692, John Woodward, sin ser consciente de ello, estaba introduciendo al mundo una nueva tecnología. Se desconoce su grado académico, pero se sabe que no asistió a la universidad, aunque llegó a ser profesor. Era un médico práctico interesado en diversas ciencias como la paleontología, geología, mineralogía y botánica, en las cuales dejó huella. Realizó unexperimento utilizando agua del parque mezclada con tierra, agua tal como la extrajo y agua destilada. Observó el crecimiento de plantas como menta, papa y arveja en estos diferentes medios, concluyendo que las plantas se alimentan del agua y de los elementos disueltos en ella, prescindiendo así del suelo para su cultivo. Este experimento dio nacimiento a la hidroponía (ARANO, 2007).

Posteriormente, esta técnica fue adoptada por el gobierno estadounidense para proveer alimentos a las tropas en el Pacífico debido a la falta de tierra cultivable y la dificultad de trans- portar alimentos frescos. El verdadero éxito de la hidroponía se manifestó después de la Segunda Guerra Mundial, y su popularidad creció significativamente en la segunda mitad del siglo XX con los avances tecnológicos. La técnica ganó aceptación especialmente en áreas conescasez de suelo fértil, ya que asegura un suministro constante de agua y nutrientes para las plantas y elimina la necesidad de usar pesticidas en el suelo(Redondo, 2023).

En los últimos años, la hidroponía ha ganado popularidad globalmente debido al aumento de

la demanda agrícola y la falta de suelo fértil, especialmente en áreas densamente pobladas como Asia. Países como Japón y Estados Unidos han avanzado significativamente en esta técnica, estableciendo grandes granjas verticales y comerciales para cultivos como lechugas y tomates. Israel y Holanda también destacan por sus innovadoras aplicaciones hidropónicas, mejorando la producción y calidad de los cultivos. En América Latina, México se ha convertido en un líder en hidroponía, aprovechando esta técnica para superar desafíos agrícolas tradicionales. Estos desarrollos reflejan un esfuerzo mundial por adoptar métodos sostenibles yeficientes para garantizar la seguridad alimentaria (Caldas et al., 2019).

2.2.2 Forrajes Verdes Hidrópicos

Los forrajes hidropónicos ofrecen múltiples ventajas, destacándose su capacidad de producir alimentos saludables y libres de pesticidas para animales domésticos. ya que de acuerdo a (Núñez & Guerrero, 2021), estos forrajes se desarrollan en sistemas de cultivo sin suelo, alimentados por soluciones nutritivas que aseguran el crecimiento adecuado de las plantas. Esta técnica permite el uso eficiente del agua y otros recursos, siendo una opción económicamenteviable para pequeños y medianos productores rurales. Además, los forrajes hidropónicos pueden ser cultivados en estructuras simples o complejas, aprovechando espacios como terrazas osuelos infértiles, y son particularmente útiles en tiempos de escasez de alimentos.

Los forrajes hidropónicos son una solución efectiva ante los desafíos del cambio climático, ya que permiten la producción de alimentos de alta calidad en condiciones controladas y con un uso eficiente del agua. Esta técnica de cultivo es especialmente útil en regiones afectadas por fenómenos adversos como sequías prolongadas y lluvias torrenciales, que limitan la producción de pasturas tradicionales. Además, los forrajes hidropónicos reducen la

dependencia de fertilizantes y pesticidas químicos, ofreciendo un recurso nutritivo y sostenible para la alimentación animal durante todo el año, incluso en épocas de escasez (J. Castillo & Núñez, 2019).

2.2.3 Importancia de la Agricultura Hidropónica en la crianza de cuyes.

Los forrajes hidropónicos ofrecen numerosos beneficios para los animales. Este tipo de forraje, producido a partir de la germinación de semillas en un ambiente controlado, asegura una fuente constante y nutritiva de alimento, independiente de las condiciones climáticas externas. Además, los forrajes hidropónicos son ricos en proteínas y otros nutrientes esenciales,lo que mejora la salud y el rendimiento de los animales. La producción de forraje hidropónico también es eficiente en el uso del agua, lo cual es crucial en áreas afectadas por sequías.

Esta técnica innovadora permite a los agricultores mantener una producción de alta calidad y regularidad, proporcionando a los animales un alimento seguro y saludable durante todo el año (Baye et al., 2024).

El Forraje Verde Hidropónico (FVH) es una opción alimenticia eficiente para cuyes, destacándose por su capacidad para aumentar la producción de crías anualmente y reducir costos dealimentación, así como las necesidades de agua y vitamina C. Este forraje se produce median-te la germinación controlada de granos durante un período de 10 a 12 días, aunque algunos lo cosechan hasta los 15 días, a pesar de que estudios indican que el máximo valor nutritivo se obtiene al día 12. El FVH alcanza una altura de 20 a 30 cm y es altamente nutritivo debido a su proceso de germinación. Su cultivo resulta en una biomasa significativa a partir de semillas de cereales, lo que lo convierte en una opción sostenible y beneficiosa para la producciónanimal (Yanchaliquin, 2022).

Dentro de los beneficios se destacan los obtenidos en el estudio de (Bonifaz et al., 2024), la digestibilidad aparente de fibra cruda y extracto etéreo en cuyes mostró ser significativamente influenciada por los niveles de azufre. Además, se estudiaron las asociaciones entre guisantes y cebada y trigo, concluyendo que los rendimientos de forraje verde y los porcentajes de materia seca y orgánica fueron similares entre las asociaciones, aunque estos porcentajes fueron afectados por la proporción leguminosa/gramínea. Los factores clave para la producción de FVH incluyen el genotipo y el momento de la cosecha, lo cual también influye en la calidad del forraje y el estado de desarrollo, permitiendo seleccionar los materiales con mayor potencial de rendimiento.

2.2.4 Tipos de forrajes para Hidroponía

La producción de forrajes puede realizarse a partir de diversas variedades de semillas; sin embargo, su rendimiento está condicionado por factores regionales y climáticos. En el presente proyecto de grado se analizan las especies forrajeras más adecuadas para las condiciones específicas de la zona de estudio, evaluando su potencial productivo y adaptabilidad.

2.2.4.1 Avena

La avena es una planta herbácea anual de la familia de las gramíneas, que destaca por su sistema radicular profundo y abundante. Sus tallos son gruesos y rectos, aunque no muy resistentes al viento, y están compuestos por varios entrenudos que terminan en nudos prominentes. Las hojas son planas, largas y estrechas, de color verde oscuro y textura áspera, con nervios paralelos bien marcados. Esta gramínea crece erecta y puede alcanzar hasta 1.5 metrosde altura, variando según la especie, y presenta inflorescencias en panícula. Su primera

cosecha se realiza aproximadamente a los 110 días de la siembra. El forraje verde hidropónico de avena, a los 10 días de cosecha, contiene un 35.49 % de materia seca (Yanchaliquin, 2022).

En la **Tabla 2** se puede notar una comparativa nutricional de este producto a los distintos días de cosecha.

Tabla 2

Aporte nutricional del forraje verde hidropónico de avena.

Constituyente (%)	**Aporte Nutricional 10 días**	**Aporte Nutricional 12 días**
Proteína Cruda	14.1	13.0
Fibra Detergente Neutra	45.7	43.0
Fibra Detergente Ácida	20.1	21.5
Lignina	2.6	3.4
Celulosa	20.8	20.7
Hemicelulosa	25.6	21.4
Extracto Etéreo	6.8	6.6
Energía Metabolizable	2200	2000

Nota: *Tabla informativa obtenida de (Yanchaliquin, 2022).*

2.2.4.2 Cebada

De acuerdo a (Hordeum vulgare L.), la cebada es un cereal ampliamente utilizado en la producción de forraje verde hidropónico debido a su alto contenido de materia seca y valor nutricional como se evidencia en la **Tabla 3**. En el sistema hidropónico, la cebada se cultiva

en un medio libre de suelo, utilizando soluciones nutritivas que proporcionan los elementos esenciales para su crecimiento. Este método de cultivo permite producir forraje fresco, rico en vitaminas y minerales, con un alto contenido de humedad, lo que lo hace ideal para la alimentación de animales como cuyes, ovinos y caprinos. La producción hidropónica de cebada es eficiente en el uso de agua y espacio, y puede seruna alternativa viable en regiones con condiciones climáticas desfavorables para la agriculturatradicional (Gómez, 2012).

Tabla 3

Constituyentes porcentuales de las diferentes partes de la cebada.

Constituyente (%)	Raíces	Tallos	Hojas	Total
Proteína cruda	12.19	27.18	35.28	16.02
Grasa	5.68	4.55	3.77	5.37
Fibra cruda	10.29	26.32	21.50	12.94
ELN	68.29	36.78	34.66	62.63
Ceniza	2.56	5.17	4.80	3.03
NDT	84.03	61.29	76.26	80.91

Nota: *Tabla basada en el trabajo de (Yanchaliquin, 2022).*

2.2.4.3 Trigo

El trigo (Triticum aestivum L.) es un cereal fundamental en la alimentación humana y animal, originario de las regiones asiáticas. En Ecuador, su cultivo se introdujo en 1535 y actualmente se cultiva principalmente en la Sierra a altitudes que van desde los 1800 hasta los 3500 metros. El trigo es esencial debido a su capacidad para formar pan voluminoso gracias al gluten presente en su harina. Este cereal no solo es una fuente importante de carbohidratos, sino que también aporta proteínas, vitaminas y minerales, los mismos que se muestran en la

Tabla 4, esenciales para la dieta humana y animal. Además,su cultivo es relevante en la rotación de cultivos y en la agricultura sostenible, donde las bacterias diazótrofas pueden jugar un papel crucial en la biofertilización, mejorando la eficiencia del uso del nitrógeno y reduciendo la dependencia de fertilizantes químicos (Cevallos, 2021).

Tabla 4

Valores nutricionales del trigo por 100g

Nutriente	**Cantidad por 100g**
Proteínas	12g
Grasas	1.7g
Almidón	70 %
Nitrógeno	2-4 % (verano), 1.75-3 % (invierno)
Fósforo	0.20-0.50 %
Potasio	1.50-3 %
Calcio	0.20-0.50 %
Magnesio	0.15-0.50 %
Azufre	0.25-0.50 %
Hierro	25-100 ppm
Manganeso	25-100 ppm
Boro	6-10 ppm
Cobre	5-25 ppm
Zinc	15-70 ppm
Molibdeno	0.09-0.18 ppm

Nota: *En la tabla se analizan las partes nutricionales del trigo de acuerdo a (Cevallos, 2021).*

2.2.5 Estudios comparativos de métodos tradicionales y modernos en la Producción de Forrajes.

Los métodos tradicionales de producción de forrajes incluyen el cultivo en suelo y el uso de técnicas convencionales de riego y fertilización. Estos métodos han sido utilizados durante siglos y se basan en prácticas agronómicas que aprovechan los ciclos naturales de crecimiento de las plantas.

En Rusia, por ejemplo, los cultivos forrajeros desempeñan un papel crucial como fuente de alimento para animales y como factor principal en la mejora de la fertilidad del suelo. Estos cultivos son esenciales para combatir la erosión y pueden crecer en suelos problemáticos, como los salinos y áridos. Sin embargo, estos métodos tradicionales enfrentan desafíos significativos debido al cambio climático y la necesidad de adaptar las variedades de cultivos forrajeros a nuevas condiciones ambientales (Kostenko et al., 2021).

Los métodos modernos incluyen técnicas como la hidroponía, la aeroponía y el uso de tecnologías de Internet de las Cosas (IoT) y computación en el borde para optimizar el cultivo de forrajes. Estas técnicas permiten un uso más eficiente del espacio y los recursos, produciendo más forraje en áreas más pequeñas y con menor consumo de agua.

Un ejemplo destacado es el sistema HydroIoT, que combina IoT, computación en el borde y visión por computadora para la producción de forraje en sistemas hidropónicos. Este sistema ha demostrado una precisión del 95% en la estimación de la producción de forraje y utiliza significativamente menos agua que los métodos tradicionales, haciendo que sea una opción viable en regiones con escasez de agua (Salvi et al., 2021).

Una de las técnicas hidropónicas más destacadas es la hidroponía recirculante, que utiliza un

sistema de agua en ciclo cerrado para nutrir las plantas. Este método no solo reduce el consumo de agua, sino que también permite reutilizar otros materiales de soporte, haciendo que el proceso sea más económico y sostenible. La hidroponía recirculante es especialmente beneficiosa en entornos urbanos donde el espacio es limitado y puede implementarse fácilmente en áreas pequeñas, ofreciendo una producción continua y de alta calidad durante todo el año (Albuja et al., 2021).

Además del contenido proteico, la irrigación con levaduras también afecta positivamente el contenido de electrólitos en el forraje, incluyendo compuestos como potasio, sodio y cloro. Estos incrementos son beneficiosos para la salud del ganado, mejorando la calidad nutricional del forraje y contribuyendo a un mejor rendimiento energético. Las ventajas de los sistemas hidropónicos, como el uso eficiente del agua y la mayor captación de nutrientes, destacan la superioridad de estos métodos modernos frente a los tradicionales (Bedolla et al., 2015).

2.3 El Cuy

Conocido también como cuye, curi, conejillo de indias, entre otros nombres, es un animal nocturno, inofensivo, nervioso y sensible al frío. Nacen con los ojos abiertos y cubiertos de pelo, pudiendo caminar y comer por su cuenta poco después de nacer. A la semana de edad, duplican su peso gracias a la nutritiva leche materna. Su peso al nacer varía según la nutrición y tamaño de la camada, y tienen una esperanza de vida de aproximadamente 8 años, aunque su explotación es óptima hasta los 18 meses debido a la disminución del rendimiento con la edad. Adaptados a una variedad de alimentos, desde desperdicios de cocina hasta forra jes y concentrados, la alimentación es crucial para su crianza y calidad (Castro, 2002).

Sistema de crianza familiar-comercial: Genera ingresos económicos y empleo en zonas rurales, ayudando a reducir la migración a las ciudades. Este sistema requiere una organización familiar, infraestructura adecuada, y una buena alimentación y manejo sanitario. Predomina el cuy mestizo, una mezcla de cuy mejorado y criollo, que produce buenos resultados de producción. Con 150 cuyes reproductores, este sistema puede generar 900 cuyes para la venta (Larrea, 2022).

Sistema de crianza comercial: Es el menos difundido pero el más tecnificado, con manejo de cuyes de líneas mejoradas que son más prolíficos y eficientes en la conversión alimenticia. Utiliza instalaciones adecuadas y organizadas, y puede incluir áreas de cultivo de forrajes y dietas balanceadas para mejorar los índices productivos (Larrea, 2022).

2.4 Sistema IoT

Hoy en día, una gran cantidad de objetos están conectados a la red mediante dispositivos electrónicos, lo que da origen al término Internet de las Cosas (IoT). Esta tecnología puede entenderse como un conjunto de métodos y técnicas que incluyen procesos como la identificación de objetos, funciones de red, inteligencia y capacidad de interacción autónoma. También puede definirse como un sistema que emplea sensores ubicuos para vincular el mundo físico con Internet(Al-Emran et al., 2020).

2.4.1 Composición de un Sistema IoT

Un sistema IoT típico consta de varios componentes clave que trabajan juntos para recopilar, procesar y transmitir datos. Estos componentes incluyen dispositivos IoT, pasarelas (gateways), la plataforma de habilitación de aplicaciones IoT (AEP), y componentes de análisis y almacenamiento de datos (Srdjan & Carrez, 2014).

2.4.1.1 Dispositivos IoT

Los dispositivos IoT pueden incluir sensores y actuadores que recopilan datos del entorno y realizan acciones basadas en esos datos. Estos dispositivos están equipados con capacidades de comunicación para transmitir datos a través de redes(Srdjan & Carrez, 2014).

2.4.1.2 Pasarelas (Gateways)

Las pasarelas son cruciales para conectar dispositivos IoT a la red. Actúan como intermediarios que permiten la comunicación entre dispositivos IoT y la nube o servidores backend. Las pasarelas pueden realizar funciones de filtrado, agregación y procesamiento preliminar de datos antes de enviarlos a sistemas más centralizados (Srdjan & Carrez, 2014).

2.4.1.3 Plataforma de habilitación de aplicaciones IoT (AEP)

La AEP proporciona un conjunto de servicios comunes que abstraen la lógica específica de la aplicación. Esto incluye la gestión de dispositivos, la conectividad, y funciones básicas de la aplicación. Al construir aplicaciones sobre una AEP, se facilita la reutilización y se reduce el tiempo de desarrollo(Arumugam et al., 2020).

2.4.1.4 Componentes de análisis y almacenamiento de datos

Estos componentes incluyen bases de datos y herramientas de análisis que almacenan y procesan los datos recopilados por los dispositivos IoT. El análisis de estos datos puede proporcionar conocimientos valiosos y soportar la toma de decisiones en tiempo real. La capacidad de estos componentes para manejar grandes volúmenes de datos es crucial para el éxito de cualquier implementación IoT(Arumugam et al., 2020).

2.4.2 Funcionamiento básico de un Sistema IoT

El funcionamiento de un sistema IoT implica varios pasos clave de los que podemos tener:

2.4.2.1 Recopilación de datos

Los dispositivos IoT, equipados con sensores, recopilan datos del entorno físico. Estos datos pueden incluir temperatura, humedad, movimiento, entre otros parámetros relevantes según la aplicación(Arumugam et al., 2020).

2.4.2.2 Transmisión de datos

Los datos recopilados son transmitidos a través de redes locales a las pasarelas, las cuales a su vez envían los datos a sistemas backend o a la nube. Este proceso puede involucrar múltiples protocolos de comunicación dependiendo de la infraestructura y las necesidades de la aplicación(Srdjan & Carrez, 2014).

2.4.2.3 Procesamiento y almacenamiento de datos

Una vez que los datos llegan a la AEP, se procesan para su análisis. Esto puede incluir la limpieza de datos, la agregación y la transformación para hacerlos adecuados para su análisis posterior. Los datos procesados se almacenan en bases de datos diseñadas para manejar grandes volúmenes de información(Srdjan & Carrez, 2014).

2.4.2.4 Análisis de Datos

El análisis de datos IoT puede involucrar técnicas de inteligencia artificial y aprendizaje automático para detectar patrones, predecir comportamientos futuros y proporcionar recomendaciones. Este análisis puede realizarse tanto en la nube como en el borde (edge) de la red para minimizar la latencia y mejorar la eficiencia(Arumugam et al., 2020).

2.4.2.5 Toma de Decisiones y Actuación

Los resultados del análisis se utilizan para tomar decisiones automatizadas o asistidas por humanos. Estas decisiones pueden desencadenar acciones a través de actuadores en el sistema

IoT, cerrando así el ciclo de recolección y uso de datos(Arumugam et al., 2020).

2.4.3 Dispositivos en sistemas IoT

La implementación efectiva del concepto de Internet de las Cosas (IoT) en escenarios reales ha sido posible gracias al progreso paralelo de las tecnologías subyacentes. En esta sección se describen estas tecnologías, con el propósito de brindar una perspectiva general sobre su función y relevancia dentro de los sistemas IoT.

2.4.3.1 Sensores

Con los avances más recientes en ciencia y tecnología, especialmente en inteligencia artificial (IA) y aprendizaje automático, la monitorización ambiental se ha convertido en un sistema de monitorización ambiental inteligente (SEM, por sus siglas en inglés), porque la tecnología ha permitido que los métodos de EM monitoreen los factores que impactan el ambiente con mayor precisión, con un control óptimo de la contaminación y otros efectos indeseables. Las redes inalámbricas o redes de sensores inalámbricos (WSN, por sus siglas en inglés) comprenden sensores modernos que operan con métodos de monitoreo y control basados en IA. Los dispositivos de internet de las cosas (IoT) se emplean en las WSN para una gestión eficaz de residuos, marcado de vehículos, control de temperatura y control de la contaminación (Ullo & Sinha, 2020).

Una de las aplicaciones que podemos tener está de acuerdo a los sistemas de gestión avícolas desarrollados por inteligente incluyen tecnologías de agricultura de precisión (PLF, por sus siglas en inglés) como sensores inteligentes, automatización de procesos agrícolas y plataformas de toma de decisiones basadas en datos. Muchas nuevas tecnologías tendrán grandes implicaciones para la producción avícola en las áreas del ambiente del gallinero,

bienestar de las aves, alimentación de precisión y detección rápida de enfermedades infecciosas. A medida que los sensores inteligentes recopilan datos en tiempo real sobre una variedad de parámetros de las operaciones avícolas(Astill et al., 2020).

2.4.3.2 Actuadores

Los actuadores son dispositivos esenciales en sistemas automatizados que convierten señales eléctricas en movimiento mecánico. Su uso es crucial en diversas aplicaciones, desde la automatización industrial hasta la agricultura de precisión. Según el documento "Development of Automated Nutrient Composition Control for Fertigation", los actuadores, junto con sensores y controladores, forman la base de los sistemas inteligentes que permiten la dosificación precisa de nutrientes en sistemas de fertirrigación. Los actuadores, como las bombas peristálticas, son controlados por microcontroladores que responden a parámetros como el pH y la conductividad eléctrica para ajustar la cantidad de nutrientes suministrados a las plantas.

2.4.3.3 Controladores

En los sistemas de telecomunicaciones, se utilizan microcontroladores, en este caso Arduino, para la monitorización en tiempo real de la gestión adaptativa y la supervisión de las fuentes de alimentación híbridas, de acuerdo a la información de (Khujamatov et al., 2019).

Como el trabajo desarrollado por (Belhakimi et al., 2023), en donde logran simplificar las tareas agrícolas y hacerlas más amigables para los agricultores, controlando la dosificación de los actuadores para optimizar las soluciones de fertilización sin causar daño a las plantas.

2.4.3.4 Fuentes de Alimentación

Según (Khujamatov et al., 2019), el suministro de energía a las instalaciones de telecomunicaciones de acuerdo con los requisitos de consumo de energía se lleva a cabo a

través del sistema de gestión de energía híbrida adaptativa.

2.5 Beneficios del uso de tecnologías IoT en la agricultura.

En la agricultura, la aplicación de IoT implica el uso de estastecnologías para empoderar a los agricultores con herramientas de decisión y tecnologías de automatización, integrando productos, conocimientos y servicios para mejorar la productividad, calidad y ganancias (Loza, 2021).

Los beneficios del uso de IoT en la agricultura de acuerdo a (Loza, 2021), incluyen:

- Disminución de desperdicios
- Mejor gestión de recursos
- Operatividad más eficiente
- Promover la agricultura comunitaria
- Mejorar la seguridad alimentaria
- Reducir costos operativos

El IoT en la agricultura tiene un impacto significativo al permitir la recopilación y análisisde datos en tiempo real para maximizar los rendimientos, minimizar el impacto ambiental y reducir costos. Además, en la industria, el IoT mejora la eficiencia de los procesos productivos mediante la optimización de la cadena de suministro y el monitoreo continuo de los activos (Llamas, 2022).

2.6 Algoritmos de inteligencia artificial aplicados a la agricultura.

La aplicación de algoritmos de inteligencia artificial en la agricultura ha demostrado ser una herramienta valiosa en diversas áreas, como el monitoreo de suelo y cultivos, el análisis predictivo y la robótica agrícola. Los sensores y el muestreo de suelo permiten recopilar datos que se utilizan en sistemas de gestión agrícola para investigaciones y análisis adicionales, facilitando la predicción de temperatura, humedad del suelo, rendimiento de cultivos y

enfermedades de las plantas(Elbasi et al., 2023).

En el caso específico de la detección de hongos Macrolepiota Procera como lo demuestra el estudio hecho por (Singh & Singh, 2023), los algoritmos de aprendizaje automático han mostrado una precisión significativa, optimizando la clasificación y monitoreo de especies dentro de los ecosistemas agrícolas y mejorando la toma de decisiones y la gestión de recursos(Singh & Singh, 2023).

La agricultura de precisión, impulsada por la inteligencia artificial, ha mejorado la eficacia y el rendimiento de las técnicas agrícolas convencionales, proporcionando beneficios en términos de calidad de los cultivos, rentabilidad, sostenibilidad y protección ambiental. Monitorear grandes campos y medir la densidad de cultivos por metro cuadrado permite predecir el rendimiento, promover la fertilización sostenible y mejorar el rendimiento general del campo(Singh & Singh, 2023).

La integración de la IA y el Internet de las Cosas (IoT) en la agricultura ha revolucionado las prácticas tradicionales, permitiendo a los agricultores optimizar sus operaciones, minimizar el desperdicio y aumentar la eficiencia general. Utilizando datos en tiempo real y análisis predictivos, los agricultores pueden responder rápidamente a las condiciones cambiantes y tomar medidas proactivas para adaptarse a los impactos del cambio climático(El Ghati et al., 2023).

Como ejemplo tenemos la aplicación de un sistema experto de rotación de cultivos basado en IA, diseñado para la aplicabilidad global, incorpora el principio de 'usuario en el bucle' (UIL) para considerar las aplicaciones regionales específicas y evitar el uso incorrecto de datos recopilados en otras regiones. Este enfoque garantiza que el asesoramiento experto sea relevante y preciso según las condiciones locales (Schoning & Richter, 2021).

En conjunto, estas tecnologías avanzadas están transformando la agricultura al proporcionar

herramientas sofisticadas para la gestión eficiente de recursos, la mejora del rendimiento de los cultivos y la promoción de prácticas sostenibles.

Dentro de estos algoritmos podemos encontrar los distintos algoritmos de aprendizaje utilizados en el control ambiental:

- **Árboles de Decisión**, uno de los algoritmos más conocidos y populares en el aprendizaje automático debido a su facilidad de interpretación y capacidad para manejar tanto variables categóricas como continuas. Se pueden emplear para predecir la necesidad de riego en sistemas agrícolas, basándose en variables como la humedad del suelo y la radiación(Bonilla, 2023).
- **Redes Neuronales Artificiales,** son modelos computacionales inspirados en el cerebro humano, capaces de aprender y generalizar patrones complejos a partir de grandes conjuntos de datos. Un estudio detalló cómo se recolectan datos de sensores de una red de sensores inalámbricos (WSN) y se utilizan para entrenar una red neuronal, la cual luego predice el estado del riego basado en la humedad del suelo y la radiación UV (Bonilla, 2023).
- **Máquinas de Soporte Vectorial (SVM),** son algoritmos de aprendizaje supervisado diseñados para tareas de clasificación y regresión. La idea fundamental de las SVM es encontrar un hiperplano óptimo que maximice el margen entre las diferentes clases en los datos de entrenamiento, lo que se traduce en una mejor capacidad de generalización al clasificar nuevos datos (Galindo et al., 2020).
- **Algoritmos de Clustering,** es una técnica no jerárquica empleada para clasificar observaciones en k grupos. Cada elemento se asigna al grupo cuyo centroide esté más cercano. El algoritmo actualiza iterativamente los grupos para minimizar la variación entre sus elementos.El algoritmo básico de k-means se utiliza con la métrica

euclidiana para determinar la distancia entre los elementos y los centros de los grupos(Stolz et al., 2020).

- **Agglomerative Hierarchical Clustering,** Este construye una jerarquía de clusters utilizando un enfoque de fusión de grupos. Comienza considerando cada punto de datos como un cluster individual y luego fusiona los clusters cercanos en cada paso. La estructura jerárquica resultante se puede visualizar como un dendrograma, facilitando la interpretación de los resultados. Sin embargo, este algoritmo puede ser computacionalmente costoso en grandes conjuntos de datos y puede ser sensible a la elección de la métrica de distancia y el método de fusión (Alvarez & Noguera, 2023).
- **Gaussian Mixture Models (GMM),** estos modelos suponen que los datos se generan a partir de una mezcla de distribuciones gaussianas. Utilizan el algoritmo de Expectation-Maximization (EM) para estimar los parámetros de las distribuciones y asignar probabilidades a cada punto de datos. GMM es útil cuando los clusters tienen distribuciones complejas y puede ser más flexible que K-Means. Sin embargo, puede tener dificultades con clusters superpuestos y su rendimiento puede verse afectado por la elección del número de componentes de mezcla (Alvarez & Noguera, 2023).

2.7 Sistemas de comunicación y protocolos en IoT.

El Internet de las cosas (IoT) forma parte de nuestra vida cotidiana y ya está presente en los dispositivos electrónicos de consumo que utilizamos, los dispositivos portátiles, los controladores industriales, los sistemas agrícolas, las estaciones de monitoreo meteorológico y muchos más. La información intercambiada en IoT puede ser relativamente pequeña, pero el ancho de banda y la energía utilizados por estos dispositivos se vuelven significativos a

medida que aumenta su número. Por lo tanto, es importante lograr la comunicación más eficiente entre los dispositivos involucrados. Un factor clave en esto es la selección de protocolos de comunicación apropiados que mejor se adapten a las necesidades de las aplicaciones(Kaskatiiski & Boyanov, 2021).

El desarrollo y la optimización de protocolos de comunicación energéticamente eficientes específicamente adaptados para dispositivos IoT es el foco de esta investigación. La proliferación de dispositivos interoperables ha generado preocupación en torno al consumo de energía, lo cual se ha convertido en una preocupación crucial en el período contemporáneo(Rahamathunnisa et al., 2024).

La IoT utiliza varias tecnologías de comunicación que se implementan en diferentes niveles de redes IoT, incluidas las redes de área personal (PAN), redes de área local (LAN), redes de área metropolitana (MAN) y redes de área amplia (WAN). Estos protocolos de comunicación tienen antecedentes técnicos únicos y áreas de interés específicas dentro del dominio de IoT(Balci & Yildiz, 2020).

Con la proliferación de dispositivos físicos conectados a Internet, garantizar la seguridad se ha convertido en un desafío crítico para el IoT, particularmente porque estos dispositivos manejan datos sensibles. Debido a la naturaleza abierta de Internet, los dispositivos conectados son vulnerables a ataques y explotación, y los sistemas IoT no seguros pueden tener consecuencias catastróficas, lo que representa un riesgo para la seguridad tanto del usuario como del objeto inteligente(Bouziane et al., 2023).

La prosperidad de IoT solo puede lograrse cuando existe una buena seguridad para los objetos y las redes de comunicación utilizadas. Es esencial implementar una política de seguridad que evite que cualquier objeto malicioso o no autorizado acceda a los sistemas IoT, lea sus datos o los modifique(Kabore, n.d.).

2.8 Plataformas de Software para Monitoreo

Una plataforma de monitoreo IoT es una capa de software middleware esencial dentro del ecosistema del Internet de las Cosas que permite la gestión, supervisión, análisis y operación de dispositivos, conectividad, y datos en entornos heterogéneos y dinámicos(Saverio, 2016). Dado el alto número de soluciones disponibles en el mercado, estas plataformas varían significativamente en los servicios que ofrecen, lo que genera un entorno rico pero complejo para los adoptantes. Por ello, se hace necesario contar con modelos de evaluación que orienten la selección de plataformas según indicadores tecnológicos y comerciales(Saverio, 2016).

De las distintas plataformas existentes se mencionan y describen las siguientes plataformas:

- **ThingSpeak**, es una plataforma de datos abierta para Internet de las Cosas (IoT) que permite la recopilación, almacenamiento y análisis de datos en tiempo real a través de una conexión HTTP habilitada por Internet. ThingSpeak es responsable de actualizar cada registro de datos que los usuarios pueden enviar y recibir de sensores, aplicaciones de estado y aplicaciones de seguimiento de ubicación(Chandrasekaran et al., 2023).
- **Node-Red, Node-RED** es una herramienta de desarrollo visual basada en flujos, creada por IBM para conectar dispositivos, servicios web y APIs dentro del Internet de las Cosas (IoT). Funciona sobre Node.js y permite crear, editar y ejecutar flujos mediante una interfaz web intuitiva. Los flujos se guardan en formato JSON, facilitando su reutilización y portabilidad. Gracias a su arquitectura modular y compatibilidad con protocolos como MQTT, Node-RED simplifica la integración,

automatización y visualización de datos en sistemas inteligentes y conectados(Chanthakit & Rattanapoka, 2018).

- **Microsoft Azure IoT**, es un servicio de comunicación en la nube que permite la conexión segura y escalable entre dispositivos IoT y soluciones. Azure IoT Hub proporciona un canal de comunicación seguro para transmitir datos desde dispositivos IoT y puede manejar millones de dispositivos al mismo tiempo. Azure IoT Central ofrece una plataforma SaaS altamente escalable que integra, analiza y mantiene activos IoT, proporcionando una solución integral para la gestión de dispositivos y datos IoT(Sreyes et al., 2022).
- **Amazon Web Services**, es una plataforma de computación en la nube poderosa con varias opciones de personalización. AWS ofrece más de 200 servicios, incluidos aprendizaje automático, inteligencia artificial, análisis y almacenamiento de máquinas. AWS proporciona una amplia gama de servicios, algunos de los cuales incluyen Elastic Compute Cloud (EC2), Simple Storage Service (S3), y Relational Database Service (Dubey & Tiwari, 2023).
- **Google Cloud IoT**, es una plataforma que permite la conexión segura y eficiente de dispositivos IoT a Google Cloud. Facilita la gestión, procesamiento y análisis de los datos generados por los dispositivos IoT, ofreciendo una infraestructura escalable y segura para el desarrollo de soluciones IoT avanzadas. Google Cloud IoT Core se integra con otros servicios de Google Cloud como BigQuery y Dataflow, permitiendo una ingesta y procesamiento de datos optimizados (Shinde et al., 2017).

2.9 Bases de Datos

Las bases de datos son herramientas fundamentales que permiten gestionar eficientemente grandes volúmenes de información en entornos modernos y competitivos. Se definen como sistemas diseñados para organizar, almacenar y recuperar datos de forma estructurada y oportuna, facilitando así la toma de decisiones dentro de una organización. Estas plataformas analizan y guían procesos empresariales, siendo el núcleo central de muchos sistemas de información. Su flexibilidad permite adaptarse a nuevas necesidades y relaciones de datos. En la actualidad, las bases de datos son esenciales en áreas como la banca, el comercio, la educación y los negocios, contribuyendo al funcionamiento eficiente de la vida diaria(Singh, 2009).

- **InfluxDB**, Es una base de datos de series temporales diseñada específicamente, desempeña un papel crucial en la gestión de datos IoT que son esencialmente marcados temporalmente y voluminosos. Esta plataforma se destaca por su capacidad para manejar datos en tiempo real y transacciones rápidas, aspectos críticos en entornos donde la analítica en tiempo real y la capacidad de transacción de datos son fundamentales(Tripathi et al., 2023).
- **MySQL**, es una base de datos relacional que organiza los datos en tablas. En el caso de la MySQL Community Server (versión 5.6), las tablas dentro de la base de datos mysql_db incluyen columnas de diferentes tipos de datos, tales como enteros, caracteres variables, booleanos y fechas. Algunas de estas columnas pueden estar cifradas utilizando el algoritmo de hash seguro (SHA y SHA-1), un proceso que es gestionado por el programa de aplicación(Vishal & Ashay, 2016).
- **PostgreSQL,** es un sistema de base de datos que utiliza un modelo de procesos en lugar de un modelo de hilos, lo que significa que, por defecto, todos los datos no se

comparten entre los procesos. La implementación de paralelismo en PostgreSQL requiere cambios significativos en la arquitectura, incluyendo trabajadores en segundo plano, memoria compartida y asignación de memoria. Desde la versión 9.4, PostgreSQL introdujo facilidades para Trabajadores Dinámicos en Segundo Plano y Memoria Compartida Dinámica, mejorando la capacidad de procesamiento paralelo(Ujibashi et al., 2015).

- **MongoDB**, Es una base de datos NoSQL orientada a documentos desarrollada por 10gen utilizando C++. Es conocida por su alto rendimiento y eficiencia. A diferencia de las bases de datos relacionales tradicionales que almacenan datos en tablas, MongoDB almacena datos como documentos BSON con esquemas dinámicos. Esto simplifica la integración de datos en ciertos tipos de aplicaciones, haciéndola más rápida y sencilla. MongoDB es especialmente adecuada para aplicaciones como sistemas de gestión de contenidos, archivado y análisis en tiempo real(Hanen et al., 2015).
- **SQLite, S**egún (Chen et al., 2010), SQLite es descrito como un excelente motor de base de datos embebido que admite la mayor parte de la sintaxis estándar SQL92, y cuya arquitectura se compone principalmente de ocho módulos organizados en tres subsistemas.El sistema de gestión de bases de datos SQLite está escrito en lenguaje C y su código fuente es abierto, y que gracias a sus ventajas de bajo consumo de recursos normalmente unos pocos cientos de kilobytes, alta velocidad, gran eficiencia y facilidad de adaptación, ha sido portado con éxito al sistema operativo embebido CLinux(Chen et al., 2010).

CAPÍTULO III: Desarrollo de la propuesta

3.1 Introducción

En esta sección inicialmente, se describe la realización del análisis situacional del sector en el que se ejecuta el proyecto, detallando las condiciones con las cuales se justifican el por qué es recomendable implementar sistemas IoT. De igual manera, se establece la ubicación física del sistema dentro de una vivienda del sector, considerando los requerimientos necesarios para integrar el prototipo con el área destinada a la crianza de los animales. En esta fase también se identifican los participantes involucrados en el desarrollo de la propuesta y se estructura un análisis detallado de los requerimientos funcionales, técnicos y de usuario.

Posteriormente, se aborda la selección de los componentes tecnológicos, entre ellos las placas electrónicas, sensores, nodos de comunicación, gateways y software de control. Además, se incorpora la fase de selección del kit de aspersión de agua, el cual desempeña un papel esencial en la hidratación controlada del FVH. Esta etapa considera tanto la compatibilidad con el sistema IoT como la adaptabilidad al entorno físico del invernadero. Para ello, se realiza una evaluación comparativa de alternativas para cada componente y se presenta la estructura general del sistema propuesto.

3.2 Descripción de la metodología

Para el desarrollo del sistema se lo realizara en base al modelo Waterfall o Cascada que de acuerdo a (Laoyan, 2024) se divide en 5 etapas fundamentales que guían el proyecto desde su inicio hasta su finalización:

- Requerimientos, es la primera y más crucial, pues en ella se recolecta toda la información necesaria para asegurar que el proyecto cumpla con los objetivos

establecidos. En esta fase se definen las dependencias, recursos y tiempos necesarios para cada etapa.

- Diseño del sistema, esta etapa incluye dos sub-etapas: una de diseño de alto nivel, donde se establece la estructura básica, y otra de diseño detallado, donde se definen los componentes técnicos específicos, como el software y hardware a utilizar.
- Implementación, en esta fase se empieza a desarrollar el producto o servicio, basándose en los requisitos y diseños previamente establecidos.
- Pruebas, a qui se verifican cada uno de los componentes para asegurarse de que cumplan con las especificaciones y estén libres de errores.
- Mantenimiento, se gestionan los posibles errores o actualizaciones necesarias una vez el proyecto esté en funcionamiento.

3.3 Requerimientos

Siguiendo la primera etapa del proceso de la metodología en cascada para el desarrollo del sistema IoT, se presenta la sección de requisitos. En esta fase, se realiza un análisis situacional del sector con el objetivo de comprender el porqué y el cómo de la implementación del sistema. Además, se lleva a cabo el desglose de los distintos requerimientos, que incluyen: identificación de los stakeholders, definición de los requerimientos del sistema, especificación de los requerimientos de hardware y software, análisis de las mejores semillas para el sistema, y selección del sistema de aspersión que se utilizará.

3.3.1 Análisis situacional

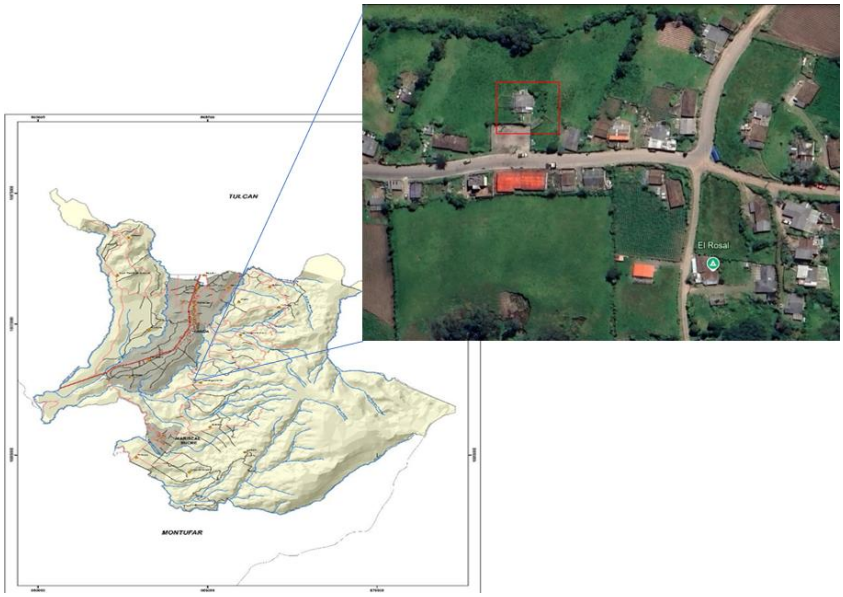
Para el análisis situacional, es necesario desglosar cada uno de los aspectos requeridos, con el fin de comprender dónde y por qué se aplicará este sistema en el sector designado. Uno de los puntos que se estudiará es precisamente esta comprensión del sector. El segundo punto corresponde a la ubicación del sistema. A continuación, se detallan estos aspectos.

3.3.1.1 Sector

El sector en el que se llevara a cabo la implementación del sistema es en el cantón San Pedro de Huaca, exactamente en una vivienda del barrio el rosal, un sector rural perteneciente al cantón como se muestra en la **Figura 3**, el mismo que es perteneciente a la señora Ana Cuasapaz, una pequeña emprendedora del sector dedicada principalmente al procesamiento de papas para venta en tiendas y que como soporte económico se dedica a la crianza de cuyes y ganado vacuno.

Figura 3

Ubicación de la vivienda



Nota: *En las imágenes se presenta el contexto geográfico del cantón y la ubicación satelital*

específica donde se implementará el sistema FVH. Esta imagen ha sido adaptada a partir de imágenes obtenidas de,(Google Earth, 2025)y (A. C. Castillo et al., 2019).

La crianza de cuyes es una actividad que enfrenta diversas limitaciones, principalmente debido a la disponibilidad restringida de forrajes adecuados para su alimentación.

La geografía del sector y el uso del suelo, enfocado en la ganadería y la agricultura tradicional de papa, limitan la producción de forrajes frescos esenciales para la crianza de estos animales. Además, las condiciones climáticas frías y variables dificultan la implementación de cultivos que necesitan condiciones ambientales térmicas e hídricas específicas. Como resultado, los pequeños productores deben depender de insumos externos o de prácticas menos eficientes, lo que incrementa los costos de producción y reduce la rentabilidad del negocio. La ausencia de tecnologías apropiadas para optimizar la producción forrajera representa un desafío que impacta directamente en la calidad de la alimentación de los cuyes y en su desarrollo productivo.

En este contexto, el proyecto se alinea con los objetivos previamente establecidos, promoviendo el uso de tecnologías emergentes en el ámbito agropecuario. La iniciativa contempla el diseño e implementación de un sistema basado en Internet de las Cosas (IoT) para la supervisión y control automatizado del crecimiento del forraje verde hidropónico (FVH).

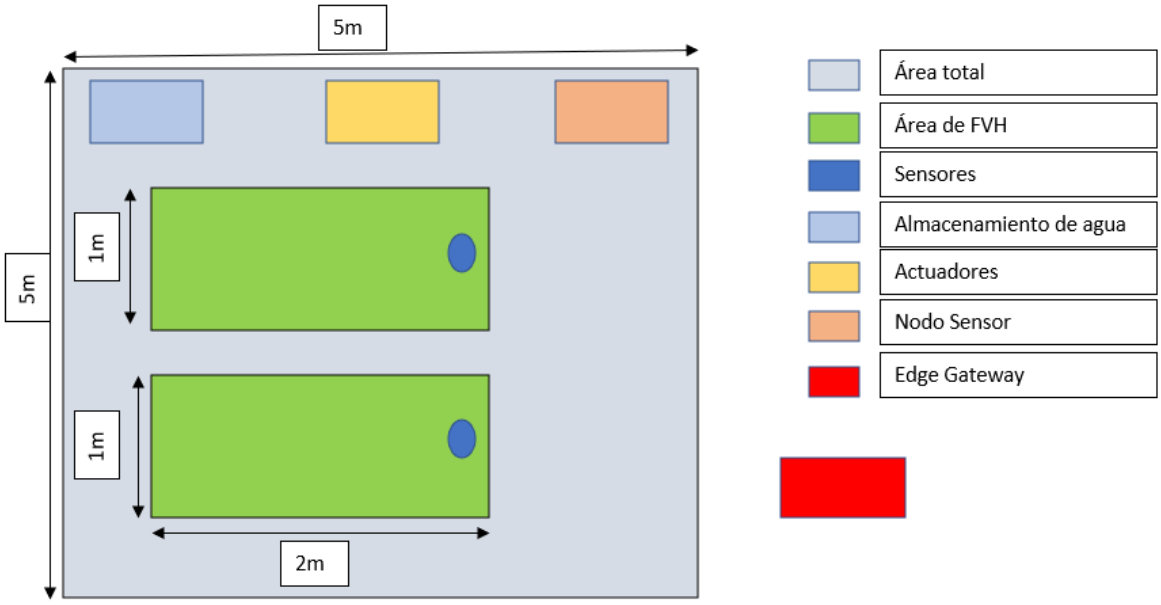
Esta solución no solo optimiza el proceso de producción, sino que también contribuye a mejorar la eficiencia y sostenibilidad de la crianza de cuyes, beneficiando especialmente a los pequeños productores del cantón San Pedro de Huaca.

3.3.1.2 Ubicación del sistema

El sistema se implementará en la vivienda de la beneficiaria, quien ha destinado un espacio integrado con el invernadero para la crianza de cuyes. El área cuenta con las dimensiones necesarias para albergar los estantes para el FVH y el sistema de medición y monitoreo, como se muestra en la **Figura 4**.

Figura 4

Esquemático del área de implementación



Nota: *Diseño de autoría propia.*

3.3.2 Análisis requerimientos

El desarrollo exitoso de un sistema IoT para la producción de (FVH) exige una formulación de requerimientos que garantice su funcionalidad y adaptabilidad a las condiciones del entorno. En esta sección se establecen los criterios técnicos, operativos y de usuario fundamentales para asegurar que el sistema cumpla con los objetivos planteados, mitigando riesgos operacionales y para lograr con ello su sostenibilidad a largo plazo.

3.3.2.1 Listado de Stakeholders

La colaboración estratégica entre todos los actores involucrados en el desarrollo e implementación del proyecto es fundamental. Desde los Docentes asesores hasta los usuarios finales, cada participante contribuye con perspectivas clave para garantizar que la iniciativa se alinee con las necesidades reales del sector agropecuario en San Pedro de Huaca. Por este motivo, a continuación, se presenta el listado de Stakeholders que participan directa o indirectamente en el proyecto, tal como se muestra en la **Tabla 5**.

Tabla 5

Listado de Stakeholders del proyecto

N°	**Involucrados**	
1	Ing. Luis Suárez	Tutor de tesis
2	Ing. Carlos Vásquez	Asesor de tesis
3	Ing. Jairo Hernández	Asesor Agrónomo
4	Erik Cuasquer	Tesista
5	Ana Cuasapaz	Productora

Nota: *Diseño de autoría propia.*

3.3.2.2 Nomenclatura

Para garantizar claridad y comprensión en la gestión del proyecto, se ha establecido un sistema de abreviaturas como se muestra en la **Tabla 6**, que permite identificar rápidamente los diferentes tipos de requerimientos. Esta estandarización facilita la comunicación entre el equipo técnico y los stakeholders, asegurando que todos los aspectos del sistema sean documentados y priorizados de manera coherente.

Tabla 6

Nomenclatura utilizada para requerimientos

Nomenclatura	Descripción
StSR	Requerimientos Stakeholders
SySR	Requerimientos del Sistema
SRSH	Requerimientos de Hardware y Software

Nota: *Diseño de autoría propia.*

3.3.2.3 Priorización de niveles

Además del análisis de las nomenclaturas de requisitos, resulta imprescindible evaluar su jerarquización según el grado de importancia como se muestra en la **Tabla 7**. Este proceso de priorización permite una selección óptima de los recursos tanto físicos como intangibles, garantizando así que cumplan con los requerimientos del usuario final durante la fase de implementación del sistema propuesto.

Tabla 7

Priorización de Requerimientos mediante niveles

Prioridad	Definición operacional
Alta	Requisitos sin los cuales el sistema no puede operar de manera efectiva.
Media	Funcionalidades importantes pero que se pueden postergar para actualizaciones a mediano plazo.
Baja	Aspectos deseables, pero pueden ser excluidos sin afectar la funcionalidad objetiva del sistema.

Nota: *Diseño de autoría propia.*

3.3.3 Requerimientos operacionales y de usuario

Una vez identificados los stakeholders del proyecto, entre los que se incluye a la productora Ana Cuasapaz, se definen los requisitos operacionales y de usuario que guiarán el diseño del sistema los cuales se describen en la **Tabla 8** respectivamente. Estas necesidades reflejan los desafíos específicos del entorno rural, como la resistencia a condiciones climáticas adversas y la facilidad de uso, asegurando que la solución sea práctica y sostenible.

Tabla 8

Requerimientos operacionales y de usuario

Requerimientos operacionales				
Numeración	**Descripción**	**Prioridad**		
		Bajo	**Media**	**Alta**
StSR 1	El sistema debe mantener condiciones ambientales óptimas para el desarrollo de FVH (Forraje Verde Hidropónico) en espacios cerrados.			X
StSR 2	El sistema debe recolectar datos de las distintas variables ambientales (temperatura, humedad, pH, luminosidad), permitiendo el entrenamiento del modelo de IA.			X
StSR 3	Debe asegurar una conexión estable entre el nodo sensor/Gateway e Internet.			X

StSR 4	El sistema debe ser energéticamente eficiente.		X	
StSR 5	Generación automática de alertas cuando los índices ambientales superen umbrales críticos predefinidos.		X	
StSR 6	Debe tener el control preciso de frecuencia y duración de riego del FVH.			X
	Requerimientos del usuario			
Numeración	**Descripción**	**Prioridad**		
		Baja	**Media**	**Alta**
StSR 7	Debe mostrar continuamente los parámetros recolectados del FVH (temperatura, humedad, pH, luminosidad) con gráficos interactivos y alertas visuales.			X
StSR 8	Todos los componentes electrónicos deben poder operar en condiciones de alta humedad (típicas en FVH).			X
StSR 9	Debe incluir recomendaciones para su uso y mantenimiento básico.		X	

Nota: *Diseño de autoría propia Fuente: Autoría.*

3.3.4 Requerimientos del sistema

Los requerimientos del sistema establecen las bases técnicas y funcionales que el sistema IoT debe cumplir para garantizar un monitoreo y control efectivo del desarrollo de FVH. Desde la estabilidad energética hasta la comunicación remota, todos estos requerimientos se muestran de manera desglosada en la **Tabla 9**, estos requisitos buscan crear una solución robusta y adaptable a las limitaciones presentes en el cantón San Pedro de Huaca.

Tabla 9

Requerimientos del sistema IoT

Requerimientos del sistema				
Uso				
Numeración	**Descripción**	**Prioridad**		
		Baja	**Media**	**Alta**
SySR 1	El sistema debe ser intuitivo y de fácil comprensión para los usuarios.			X
SySR 2	Los datos mostrados al usuario deben ser coherentes y similares a los referentes por otras fuentes de datos.			X
SySR 3	Se debe notar la constante comunicación entre los componentes del sistema con una latencia aceptable o mínima.		X	
SySR 4	El monitoreo remoto debe darse mediante una app o plataforma web.		X	
Requerimientos de funcionamiento				

SySR 5	Los protocolos de comunicación usados por el sistema deben ser 802.11 Y MQTT de acuerdo al diseño del sistema.		X
SySR 6	Envió de datos al Gateway debe ser periódico con un máximo de 15 minutos de actualización.	X	
SySR 7	Las partes del sistema deben tener un protocolo de arranque y apagado periódico.	X	
SySR 8	El algoritmo de IA de ser capaz de decidir la activación del riego basado en datos históricos.		X
SySR 9	Debe controlar los actuadores de manera correcta y estos deben dar la potencia necesaria para el sistema de FHV.		X
	Requerimientos de interfaces		
SySR10	El nodo controlador deberá tener suficiente capacidad para la conexión de sensores y actuadores.		X
SySR11	La conexión de las interfaces debe ser figa y resistente a condiciones expuestas.	X	
SySR12	La comunicación debe ser de acuerdo a los protocolos estudiados.		X

SySR13	Las placas deben ser compatibles mutuamente.	X	
	Requerimientos físicos		
SySR 14	Debe ser resistente a la exposición de humedad polvo y factores externos.		X
SySR 15	Debe tener un espacio dedicado para ubicarse acuerdo a sus dimensiones.	X	
SySR 16	Los sensores de humedad deben estar ubicados de forma estratégica en las bandejas para FVH.	X	
SySR 17	El sistema debe ser alimentado por una fuente eléctrica fija y continua.		X
	Requerimientos de modos de estado		
SySR 18	En estado activo el sistema censará el ambiente y actuará de acuerdo a los datos medidos.		X
SySR 19	En caso de fallas el sistema debe dar un aviso de errores.		X
SySR 20	Deberá constar con un sistema de reposo para ahorro energético.	X	
SySR 21	Debería ser capaz de reiniciarse después de una intermitencia eléctrica.		X

Nota: *Diseño de autoría propia.*

3.3.5 Requerimientos de la arquitectura

La arquitectura tecnológica del sistema requiere componentes de hardware resilientes y de software adaptable. Sensores capaces de proporcionar lecturas con alto nivel de precisión, interfaces programables y plataformas de análisis de datos constituyen elementos clave para lograr un que sea escalable, que responda a las necesidades actuales y futuras de la producción de FVH en entornos rurales. A continuación, mostramos los puntos de vista necesarios para nuestro caso en el sistema, tal como se ve en la **Tabla 10**.

Tabla 10

Requerimientos de la arquitectura del sistema IoT

Requerimientos de la arquitectura				
Numeración	**Descripción**	**Prioridad**		
		Baja	**Media**	**Alta**
	Requerimientos lógicos			
SRSH 1	El sistema debe establecer una comunicación bidireccional entre Gateway, nodo y sensores.			X
SRSH 2	El sistema debe incluir rutinas de calibración para cada tipo de sensor, permitiendo ajustar las mediciones con valores de referencia.			X
SRSH 3	El sistema debe usar protocolos de bajos requerimientos de procesamiento.		X	
SRSH 4	La lógica de control debe procesar los datos			

	en el servidor para decidir las acciones de los actuadores.		X
	Requerimientos de diseño		
SRSH 5	El sistema debe tener diferentes módulos de funcionamiento.		X
SRSH 6	El armazón fisco debe tener una estructura vertical para optimizar espacio.		X
SRSH 7	Debe constar con distribución de cableado y estructuración que permita el mantenimiento.		X
SRSH 8	Debe estar diseñado con un enfoque en la escalabilidad tanto de sus componentes como de su arquitectura.	X	
	Requerimientos de hardware		
SRSH 9	El nodo controlador debe tener capacidad para múltiples Entradas y salidas Analógico/Digital.		X
SRSH 10	Los sensores deben garantizar precisión en la medición de variables y permitir ser calibrados.		X
SRSH 11	El Gateway debe tener capacidad robusta de procesamiento de datos.		X
SRSH 12	Los actuadores como electroválvulas deben		

	ser de garantía y durabilidad.		X
SRSH 13	Los componentes deben tener disponibilidad en el mercado ecuatoriano y un precio asequible, o poder importarse sin restricciones.	X	
	Requerimientos de software		
SRSH 14	El Lenguaje de programación debe ser de uso libre y con flexible en su aplicación.		X
SRSH 15	Disponibilidad de librerías compatibles para los sensores y el bloque de procesamiento.		X
SRSH 16	Debe poseer un dashboard web en tiempo real o un framework para visualizar datos, estados y configurar alertas.	X	
	Requerimientos eléctricos		
SRSH 17	La alimentación de los dispositivos debe ser independiente y de un voltaje adecuado.		X
SRSH 18	La alimentación del Gateway debe ser estable de acuerdo a las especificaciones de la placa elegida.		X
SRSH 19	Todos los componentes deben tener su estructura de protección eléctrica o ser compatibles al AC.		X
SRSH 20	El cableado debe seguir su etiquetado y/o		

código de colores para diferenciar.	X

Nota: *Diseño de autoría propia.*

3.4 Diseño del sistema

Para iniciar el diseño del sistema, se procedió a la selección de los componentes óptimos, tanto de hardware como de software, considerando que estos cumplieran con los requisitos técnicos del sistema y las necesidades funcionales del usuario.

3.4.1 Selección de hardware

El proceso de selección del hardware se basó en analizar los componentes físicos que integrarán el sistema. Inicialmente, se evaluaron las características del Gateway y del nodo primario, encargados de la conexión con los sensores y actuadores. La información para este análisis se obtuvo de las hojas técnicas de varias placas, lo que permitió definir un panorama claro de las alternativas.

3.4.1.1 Selección del Nodo Sensor y Actuador.

En cuanto al nodo sensor, se presenta la siguiente tabla donde se hace notar las características encontradas de las distintas opciones de placas las mismas que se observan en la **Tabla 11**.

Tabla 11

Características de las placas para nodo sensor y actuador

Características	**Esp32 LORA**	**Arduino UNO**	**Arduino MEGA 2560**	**Esp-WROOM-32D**	**Esp8266**
Voltaje de operación	3.3 V	5 V	5 V	3.3 V	3.3 V
Corriente promedio	80 mA	50 mA	70 mA	160 mA	70 mA
Velocidad del procesador	240 MHz	16 MHz	16 MHz	240 MHz	80/160 MHz
Memoria Flash	4 MB	32 KB	256 KB	4 MB	1 MB
RAM	520 KB	2 KB	8 KB	512 KB	160 KB
Wi-Fi	Sí	No	No	Sí	Sí
Bluetooth	Sí	No	No	Sí (BLE)	No
Nº pines digitales	20+	14	54	34	17 (varía)
Precio aproximado	$8	$10	$15	$6	$4
Tamaño	5.5x2.5 cm	6.8x5.3 cm	10.2x5.3 cm	5.5x2.5 cm	3.4x2.5 cm

Nota: *Diseño de autoría propia.*

De acuerdo con los requerimientos del sistema, a continuación, se presenta en la **Tabla 12**, una comparativa de las placas con mayor potencial para el proyecto. Este análisis facilitará

la selección del equipo más adecuado, enfocándose en esta instancia en las placas que pueden cumplir la función de nodo sensor.

Tabla 12

Selección de nodo sensor y actuador

	REQUISITOS									
	SRSH 1	SRSH 3	SRSH 9	SRSH 11	SySR 18	SySR 5	SySR 6	SySR 10	SySR 20	Val
ESP32 LoRa	1	1	1	1	1	1	1	1	1	9
Arduino UNO	0	1	0	0	1	0	0	0	1	3
Arduino Mega 2560	0	1	1	1	1	0	0	0	1	5
ESP-WROOM-32D	1	1	1	1	1	1	1	1	1	9
ESP8266	1	1	0	1	1	1	1	0	1	7
Cumple “1” No cumple “0”										
Selección: La placa embebida seleccionada para este proyecto es, **ESP-WROOM-32D**										

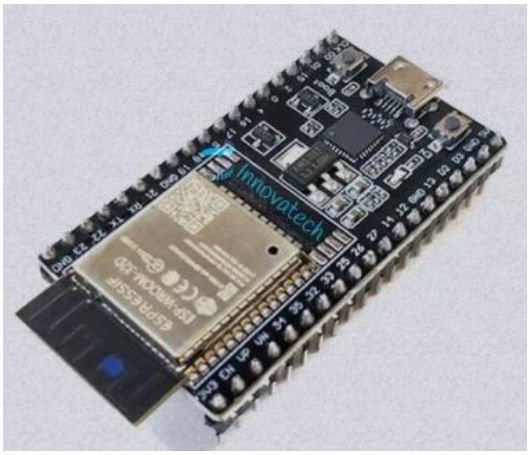
Fuente: Autoría

El análisis de las especificaciones técnicas en la **Tabla 11**, junto con los resultados de la evaluación comparativa basada en criterios positivos y negativos presentada en la **Tabla 12**, confirma que la placa **ESP-WROOM-32D** constituye la opción óptima para la función de nodo sensor y actuador. Esta decisión se fundamenta en el cumplimiento de los requisitos establecidos, su versatilidad y relación costo-beneficio. Dado que el sistema está diseñado para un entorno de corto alcance, la funcionalidad de comunicación LoRa no se consideró un

requisito, lo cual consolida la idoneidad del modelo seleccionado para los fines del proyecto.

Figura 5

Placa ESP-WROOM-32D seleccionada para nodo sensor y actuador



Nota: *Imagen de la placa ESP-WROOM-32D seleccionada como nodo sensor y actuador, imagen obtenida de (Mercado libre, 2025b).*

3.4.1.2 Selección del Edge Gateway

Siguiendo la misma metodología de evaluación definida para el nodo sensor, se procedió con la selección de la placa para Edge Gateway. Esto implicó una comparación de las especificaciones técnicas de los modelos disponibles en el mercado como se muestra en la **Tabla 13**, priorizando aquellos que ofrecen el mejor balance entre rendimiento, confiabilidad y costo además prestando especial atención a su compatibilidad con el nodo sensor seleccionado, satisfaciendo así las necesidades específicas del sistema.

Tabla 13

Características de placas para Edge Gateway

Características	**Raspberry Pi 4B**	**Raspberry Pi 5B**	**Orange Pi 5**	**Jetson Nano**	**BeagleBone Black**
CPU	Quad-core Cortex-A72 1.5 GHz	Quad-core Cortex-A76 2.4 GHz + Cortex-A55	Hexa-core RK3588 4x Cortex-A76 + 4x Cortex-A55	Quad-core Cortex-A57 1.43 GHz	ARM Cortex-A8 1 GHz
GPU	VideoCore VI	VideoCore VII	Mali G610 MP4	128-core Maxwell GPU	PowerVR SGX530
RAM	2GB, 4GB, 8GB LPDDR4X	4GB, 8GB LPDDR5	8GB, 16GB LPDDR4	4GB LPDDR4	512MB DDR3
Almacenamiento	microSD	microSD, NVMe SSD	eMMC, microSD, NVMe	microSD	microSD, eMMC
Conectividad Ethernet	Gigabit Ethernet	2.5 Gigabit Ethernet	2.5 Gigabit Ethernet	Gigabit Ethernet	10/100 Ethernet
Wi-Fi	802.11ac (Wi-Fi 5)	802.11ax (Wi-Fi 6)	802.11ax (Wi-Fi 6)	No integrado	No integrado

Bluetooth	5.0	5.0	5.0	No	4.1
Puertos USB	2x USB 3.0, 2x USB 2.0	2x USB 3.0, 2x USB 2.0	4x USB 3.0	4x USB 3.0	1x USB 2.0
Video Salida	2x micro HDMI 4K60	2x micro HDMI 4K60	2x HDMI 2.1	HDMI 2.0	HDMI
GPIO	40 pines	40 pines	40 pines	40 pines	65 pines
Alimentación	5 V DC (USB-C)	5 V DC (USB-C)	12 V DC	5 V DC	5 V DC
Dimensiones	85.6 x 56.5 mm	85.6 x 56.5 mm	90 x 65 mm	100 x 80 mm	86 x 53 mm
Precio aproximado	$35 (4GB)	$60 (4GB)	$90 (8GB)	$99	$55

Nota: *Diseño de autoría propia.*

El siguiente paso consistió en realizar un análisis comparativo permitiendo filtrar las opciones como se evidencia en la **Tabla 14**, descartando todos los modelos que no cumplían con los requisitos establecidos. De este modo, la selección se centra en el Edge Gateway que representa la solución más idónea para el proyecto.

Tabla 14

Selección de placa que funcionará como Edge Gateway

	Requisitos										
	SRSH 1	**SRSH 4**	**SRSH 11**	**SRSH 13**	**SRSH 18**	**SySR 4**	**SySR 5**	**SySR 10**	**SySR 19**	**StSR 2**	**Val.**
Raspberry Pi 4B	1	1	1	1	1	1	1	1	1	1	1
Raspberry Pi 5B	1	1	1	1	1	1	1	1	1	1	1
Orange Pi 5	1	0	1	1	1	1	1	1	1	1	9
Jetson nano	1	1	1	0	1	1	0	1	1	1	8
BeagleBone Black	1	0	0	0	1	0	0	1	1	1	5
	Cumple “1” No cumple “0”										
Selección: La placa embebida seleccionada para este proyecto es, **Raspberry Pi 5**											

Nota: *Diseño de autoría propia.*

El análisis comparativo determinó que la placa **Raspberry Pi 5,** la cual se presenta en la **Figura 6**, es la opción más idónea, al cumplir de manera integral con todos los requerimientos del sistema. Su selección se fundamenta en la necesidad de garantizar un funcionamiento robusto, una mayor capacidad de procesamiento de datos y una comunicación de alta velocidad, características que este dispositivo ofrece gracias a su potente CPU de cuatro núcleos y su interfaz de Gigabit Ethernet nativa.

Figura 6

Placa seleccionada para Edge Gateway



Nota: *Imagen obtenida en plataforma (Mercado libre, 2025c), placa Raspberry Pi 5.*

En comparación con su antecesora, la Raspberry Pi 4, si bien ambas cumplen con los requisitos mínimos establecidos, la versión 5 presenta un rendimiento significativamente superior, un mejor control térmico durante su operación y una mayor capacidad de proyección para futuras actualizaciones o la incorporación de nuevos módulos dentro del sistema.

3.4.1.3 Selección de los Sensores

Dentro de esta subsección se procede a la selección de los sensores a utilizar. Para determinar los más adecuados para el proyecto, se investigaron diversas opciones y tipologías, analizando sus características técnicas de manera individual.

3.4.1.3.1 Sensor de Temperatura

Para la selección del sensor de temperatura se establecieron criterios como la relación calidad-precio y las especificaciones técnicas de las distintas alternativas disponibles en el mercado. En la **Tabla 15** se presentan las opciones evaluadas.

Tabla 15

Características de posibles sensores de temperatura

Característica	**DS18B20**	**DHT22**	**SHT35**
Rango de temperatura	-55°C a +125°C	-40°C a +80°C	-40°C a +125°C
Precisión de temperatura	±0.5°C (de -10°C a 85°C)	±0.5°C	±0.2°C (de 0°C a 65°C)
Voltaje de operación	3.0V a 5.5V	3.3V a 6V	2.4V a 5.5V
Consumo de corriente	1mA (activo) 750 nA (reposo)	1-1.5mA (medición) 40-50 µA (reposo)	1.2 mA (medición) 0.5 µA (reposo)
Tiempo de medición	750 ms	2 segundos	8 ms
Conexión múltiple	Sí (hasta 255 en un bus)	No (1 sensor por pin)	Sí (hasta 8 en bus I2C)
Resistencia a humedad	Alta (sensor encapsulado)	Media (placa expuesta)	Alta (carcasa IP67)
Costo aproximado	$2 - $4 USD	$4 - $8 USD	$15 - $25 USD

Nota: *Diseño de autoría propia.*

La selección del sensor de temperatura se realizó en función de los requerimientos ya establecidos, como se muestra en la **Tabla 16**. Esta evaluación comparativa priorizó la precisión, confiabilidad y compatibilidad con el entorno del sistema FVH. Garantizar que el sensor cumpla con los estándares establecidos es fundamental para un control térmico exacto, factor que incide en la eficiencia del crecimiento del forraje y, en consecuencia, en el desempeño integral del sistema automatizado.

Tabla 16

Requisitos para sensor de Temperatura

	Requisitos							
	StSR 4	**StSR 8**	**SRSH 10**	**SRSH 13**	**SRSH 15**	**SRSH 17**	**SySR 16**	**Val.**
DS18B20	1	1	1	1	1	1	1	7
DHT22	1	1	1	1	1	1	1	7
SHT35	1	1	1	0	1	1	0	6
Cumple “1” No cumple “0”								
Selección: el sensor seleccionado para este proyecto es, **DHT22**								

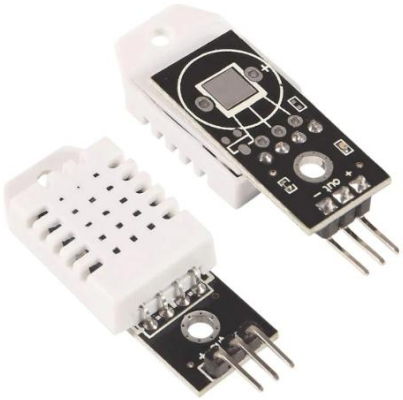
Nota: *Diseño de autoría propia.*

Aunque ambos sensores evaluados cumplen con las expectativas técnicas, el **DHT22** fue seleccionado por su idoneidad específica para medir la temperatura ambiental, parámetro fundamental en el control del microclima del sistema FVH, este sensor se le puede ver en la **Figura 7**. Esta elección se ve reforzada por su amplia disponibilidad en el mercado, lo que facilita la adquisición y el reemplazo, así como por su facilidad de implementación, agiliza

la integración y el mantenimiento del sistema de monitoreo.

Figura 7

Sensor de temperatura seleccionado es el Modelo *DTH 22*



Nota: *Imagen obtenida de(Mercado libre, 2025a), sensor DTH 22.*

3.4.1.3.2 Selección del sensor de Humedad

Para el sensor de humedad, se analizaron las necesidades específicas del proyecto FVH y se realizó una búsqueda de las alternativas disponibles en el mercado. Esta investigación incluyó la consulta de las hojas técnicas (datasheets) de cada modelo, permitiendo comparar sus características, rangos de medición, precisión y compatibilidad con el sistema. Como se documenta en la **Tabla 17**, este proceso comparativo resulto esencial para proceder de manera fundamentada a la selección del sensor más adecuado para las condiciones de monitoreo de humedad.

Tabla 17

Características de las opciones de sensor de humedad

Características	FC-28	SHT85	Capacitivo V2.0
Tipo	Resistivo	Quad-core	Hexa-core
Rango	0-100%	0-100%	0-100%
Precisión	±5-10%	±1.5%	±2-3%
Voltaje operación	3.3V - 5V	2.4V - 5.5V	3.3V - 5.5V
Salida	Analógica/Digital	Digital I²C	Analógica
Consumo	15 - 20 mA	0.6 - 1.2 mA	5 - 10 mA
Durabilidad	Baja	Alta	Alta
Costo	$3	$30	$5

Nota: *Diseño de autoría propia.*

De manera similar al caso de la temperatura, en la **Tabla 18** se detallan los requerimientos específicos establecidos para el sensor de humedad. Es fundamental que este dispositivo permita un monitoreo preciso y confiable de los niveles de humedad en la zona de producción del cultivo, ya que esta medición constituye la variable principal para la activación o desactivación automatizada del sistema de riego, asegurando así las condiciones óptimas para el desarrollo del forraje.

Tabla 18

Requisitos para selección del sensor de humedad

	REQUISITOS								
	SRSH 2	**SRSH 10**	**SRSH 13**	**SRSH 17**	**StSR 8**	**StSR 9**	**SySR 14**	**SySR 16**	**Val.**
FC-28	0	0	1	1	0	1	0	1	4
SHT85	1	1	0	1	1	0	1	1	6
Capacitive v2.0	1	1	1	1	1	1	1	1	8
Cumple “1” No cumple “0”									
Selección: el sensor seleccionado para este proyecto es, **Capacitivo v2.0**									

Nota: *Diseño de autoría propia.*

Tras el análisis comparativo, se determinó que el sensor **Capacitivo v2.0** es el que mejor se ajusta a los requerimientos del proyecto, al cumplir de mejor manera con todos los criterios técnicos definidos, tales como su precisión, durabilidad en medios húmedos y respuesta adecuada para el sistema de FVH. En la **Figura 8** se muestra el modelo específico seleccionado (HW-390) para su adquisición e integración en el sistema.

Figura 8

Sensor Capacitivo v2.0



Nota: el sensor seleccionado es el modelo capacitivo versión 2.0 imagen obtenida de (Mercado libre, 2025d)

3.4.1.3.3 Selección nodo Sensor de Luz

En cuanto al sensor de luminosidad se procedió a la evaluación de las distintas opciones disponibles en el mercado para este tipo de sensores. En la **Tabla 19** se presentan las características técnicas fundamentales de las alternativas analizadas, con el fin de seleccionar el dispositivo que mejor se adapte a los requerimientos específicos de monitoreo lumínico del sistema FVH.

Tabla 19

Características de las opciones para sensor de luminosidad

Característica	**TSL2591**	**BH1750**	**LDR**
Principio de funcionamiento	Digital (I2C)	Digital (I2C)	Analógico
¡Rango de medición	0.188 lux to 88,000 lux	1 – 65535 lux	Depende del circuito (no lineal)
Precisión	±0.96 lux	±1.0 lux	±20-30%
Longitudes de onda	IR + Visible	Visible	Amplio espectro
Interfaz	I2C	I2C	Analógica
Voltaje de operación	2.6V a 3.6V	2.4V a 3.6V	3.3V a 5V
Consumo de corriente	0.65 mA (activo) 2 µA (apagado)	0.12 mA (activo) 1 µA (apagado)	0.1 – 5 mA (depende de la luz)
Costo aproximado	$4 - $8 USD	$2 - $4 USD	$0.10 - $0.50 USD
Facilidad de uso	Media	Alta	Muy alta
Calibración requerida	Media	Baja	Alta

Nota: *Diseño de autoría propia.*

Finalmente, se definieron los requerimientos específicos para la selección del sensor de luz solar, cuya medición es determinante para monitorear el desarrollo óptimo del Forraje Verde Hidropónico (FVH). Estos criterios, enfocados en la precisión, el rango de medición y la integración con el sistema de control, se presentan en la **Tabla 20**, donde se comparan las

opciones disponibles y se fundamenta la selección final. El sensor elegido permitirá un monitoreo preciso de la intensidad lumínica.

Tabla 20

Requisitos para selección del sensor de luz solar

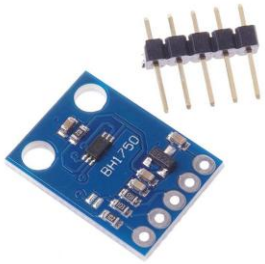
	Requisitos								
	StSR 2	**StSR 4**	**SRSH 10**	**SRSH 13**	**SRSH 15**	**SRSH 17**	**SySR 2**	**SySR 16**	**Val.**
TSL2591	1	1	1	0	1	1	1	1	7
BH1750	1	1	1	1	1	1	1	1	8
LDR	0	1	0	1	1	1	0	1	5
Cumple “1” No cumple “0”									
Selección: el sensor seleccionado para este proyecto es**, BH1750**									

Nota: *Diseño de autoría propia.*

De acuerdo con el análisis de los requerimientos, el sensor seleccionado para la medición de luminosidad es el BH1750, tal como se presenta en la **Figura 9**. Esta elección se fundamenta en su capacidad para ofrecer mediciones precisas en un rango amplio de intensidad lumínica, su interfaz de comunicación I²C que facilita la integración con el microcontrolador, y su bajo consumo energético, características que lo hacen idóneo para el monitoreo continuo en el sistema FVH.

Figura 9

Sensor seleccionado para monitorear luminosidad



Nota: *Imagen del Sensor BH1750 obtenida en plataforma (Novatronic, 2020).*

3.4.1.3.4 Selección de sensor de pH

Mantener un monitoreo constante del pH del agua en el sistema de riego hidropónico es fundamental para garantizar el desarrollo óptimo del forraje, asegurar su valor nutricional y prevenir afectaciones por enfermedades o desequilibrios químicos. Un control preciso de este parámetro permite crear un ambiente estable que favorece la absorción de nutrientes y evita condiciones que puedan inhibir el crecimiento de las plantas.

Para seleccionar el sensor más adecuado, se realizó una evaluación de las opciones disponibles en el mercado, considerando criterios como precisión, estabilidad de medición, durabilidad en medios acuosos y facilidad de calibración. El análisis comparativo de estas alternativas se presenta en la **Tabla 21**.

Tabla 21

Características sensores de pH

Característica	**PH-4502C**	**DFRobot pH Meter V1.1**	**Gravity pH Sensor V2**
Voltaje de Operación	5.0V +0.2V	5.00V	3.3V a 5.5V
Consumo de Corriente	5 - 10mA	5 - 10mA	5 - 10mA
Rango de Medición	0 - 14 pH	0 - 14 pH	0 - 14 pH
Precisión (a 25°C)	0.1 pH	0.1pH	0.1 pH
Tiempo	5 segundos	1 minuto	1 minuto
Rango de Temperatura	0 - 60 °C	0 – 60 de °C	0 - 60 °C
Tipo de Salida	Analógica	Analógica	Analógica

Nota: *Diseño de autoría propia.*

Una vez identificadas las alternativas de sensores de pH disponibles en el mercado, se procedió a realizar una evaluación comparativa estructurada en función de los requerimientos específicos del sistema FVH. Esta comparativa, presentada en la **Tabla 22**, con el fin de seleccionar el sensor más idóneo para garantizar un monitoreo confiable y preciso del pH en el agua de riego.

Tabla 22

Requisitos para selección del sensor de ph

	Requisitos							
	StSR 2	StSR 8	**SySR 9**	**SySR 1**	**SRSH 10**	SRSH 13	**SRSH 15**	**Val.**
PH-4502C	1	1	1	1	1	1	1	7
DFRobot pH Meter V1.1	1	1	1	1	1	0	1	6
Gravity pH Sensor V2	1	1	1	1	1	0	1	6
Cumple “1” No cumple “0”								
Selección: el sensor seleccionado es el, PH-4502C								

Nota: *Diseño de autoría propia.*

Como se muestra en la **Figura 10**, el sensor seleccionado es el PH-4502C. Esta selección se fundamenta en que, si bien todas las opciones evaluadas cumplen con las expectativas técnicas mínimas, el modelo PH-4502C presenta un costo significativamente menor en el mercado ecuatoriano, manteniendo al mismo tiempo un desempeño adecuado para los requerimientos de precisión y fiabilidad del sistema de monitoreo de pH en el agua de riego.

Figura 10

Selección del Sensor para pH de agua: Modelo PH-4502C



Nota: *el sensor seleccionado es el modelo PH-4502C, imagen obtenida de (Mercado libre, 2025e)*

3.4.1.4 Selección de bomba de riego

Para la selección de la bomba de riego, se realizó una investigación de mercado entre los principales proveedores del sector, evaluando modelos que cumplieran con las condiciones de potencia, presión y caudal requeridas para el sistema de aspersión. Los criterios de selección incluyeron, además de las especificaciones técnicas, el consumo energético, la disponibilidad en el mercado local y las condiciones de garantía. En la **Tabla 23** se detallan las características técnicas de las bombas preseleccionadas.

Tabla 23

Características de bomba para riego

Especificación	Paolo PC-122	Truper BOAP-1/2	Pedrollo CPM 158
Potencia	1 HP (0.75 kW)	1/2 HP (0.37 kW)	1 HP (0.75 kW)
Voltaje	110 / 220 V (Doble voltaje)	127 V (Fijo)	115 / 230 V
Corriente (Amperaje)	11 A (a 110V) / 5.5 A (a 220V)	5.2 A (a 127V)	~13.5 A (a 115V) / 7.0 A (a 230V)
Velocidad (RPM)	3400 min^{-1} (Alta velocidad)	3450 RPM	3450 RPM
Protección Térmica	SÍ	No siempre	SÍ
Ciclo de Trabajo	Servicio Continuo	50 min trabajo / 20 descanso	Servicio Continuo
Capacitor	50 µF	~20-25 µF	~35-40 µF
Caudal Máximo	600 l/min	40 l/min	90 l/min

Nota: *Diseño de autoría propia.*

Tras el análisis de especificaciones, se procedió a evaluar el cumplimiento de cada opción frente a los requerimientos funcionales y de seguridad del sistema, tal como se presenta en la **Tabla 24**.

Tabla 24

Requisitos para selección de Bomba de riego

	Requisitos							
	StSR 1	**StSR 6**	**SySR 14**	**SRSH 12**	**SRSH 13**	**SRSH 17**	**SRSH 19**	**Val**
Paolo PC-122	1	1	1	1	1	1	1	7
Truper BOAP-1/2	0	0	1	1	1	1	1	5
Pedrollo CPM 158	0	1	1	1	0	1	1	5
Cumple “1” No cumple “0”								
Selección: La bomba seleccionada para este proyecto es, **Paolo PC-122**								

Nota: *Diseño de autoría propia.*

La bomba **Paolo PC-122** resultó ser la que satisface de manera integral todos los criterios técnicos y operativos establecidos, por lo que fue seleccionada para la implementación del proyecto. Su diseño puede observarse en la **Figura 11**, la cual muestra el modelo específico que será integrado en el sistema de riego del proyecto FVH.

Figura 11

Bomba de riego modelo Paolo PC-122



***Nota**: bomba a utilizarse para suministrar caudal al sistema de aspersión, imagen de autoría.*

3.4.1.5 Selección del Contactor para el Circuito de Potencia

El contactor es el dispositivo encargado de aislar y controlar el circuito de potencia que alimenta la bomba de riego, garantizando su funcionamiento seguro sin interferir en la integridad del circuito de monitoreo y control del sistema FVH. Dada su importancia crítica para la operación y protección del sistema, se investigaron distintos tipos de contactores disponibles en el mercado. Las opciones evaluadas, considerando parámetros como capacidad de corriente, tipo de bobina y costo en el mercado, se presentan en la **Tabla 25**.

Tabla 25

Características de modelos de bomba

Característica	**Camsco C1-D2510**	**Schneider LC1D25**	**Chint NC1-1810**	**Sassin CJX2-0910**
Capacidad Máxima	25 A	25 A	18 A	9 A
Voltaje de Bobina	120V / 60Hz	120V / 60Hz	110V-120V / 60Hz	110V-120V / 60Hz
Corriente de Consumo Bobina (Inrush)	60 - 70 VA	70 VA	50 - 60 VA	45 - 55 VA
Corriente de Consumo Bobina (Hold)	7 - 10 VA	7 VA	8 - 10 VA	6 - 8 VA
Contactos Auxiliares	1 NO	1 NO + 1 NC	1 NO	1 NO
Vida Útil Eléctrica	Media-Alta	Muy Alta	Media	Estándar

Nota: *Diseño de autoría propia.*

Una vez definidas las características técnicas de cada modelo y contrastadas con los requerimientos del sistema, se procedió a evaluar las distintas opciones de contactores disponibles. Esta evaluación comparativa, presentada en la **Tabla 26**. Dicho análisis permite identificar la opción más coherente y adecuada para garantizar un control seguro y confiable del circuito de potencia de la bomba de riego.

Tabla 26

Evaluación de requerimientos para contactor

	Requisitos							
	StSR 4	**StSR 6**	**SySR 9**	**SRSH 12**	**SRSH 13**	**SRSH 17**	**SRSH 19**	**Val.**
Schneider LC1D25	1	1	1	1	0	1	1	6
Camsco C1-D2510	1	1	1	1	1	1	1	7
Chint NC1-1810	1	1	1	0	1	1	1	6
Cumple “1” No cumple “0”								
Selección: el contactor seleccionado es el, Camsco C1-D2510								

Nota: *Diseño de autoría propia.*

Tras evaluar las opciones presentadas en la **Tabla 24**, se seleccionó el contactor **Camsco C1-D2510**, Mismo que se observa en la **Figura 12**, como la solución más equilibrada para el sistema. Si bien el modelo Schneider LC1D25 presenta ventajas en durabilidad y desempeño, el contactor Camsco satisface todos los requerimientos técnicos establecidos, ofreciendo además un costo significativamente menor y una amplia disponibilidad en el mercado local, factores determinantes para la viabilidad económica y logística del proyecto.

Figura 12

Contactor seleccionado para el sistema.



***Nota**: imagen del contactor **Camsco C1-D2510**, obtenida de (Mercado libre, 2026).*

3.4.2 Selección de software

El software constituye uno de los pilares fundamentales de cualquier sistema electrónico-informático, y en particular, de un sistema IoT. Actúa como el alma del sistema, garantizando que el hardware seleccionado funcione según lo previsto en los puntos anteriores. Para ello, se considerarán aspectos esenciales como la capacidad de procesamiento tanto del nodo como del gateway, así como su compatibilidad mutua. Esto permitirá evitar errores y problemas de conexión entre placas, plataformas y lenguajes de programación, facilitando el mantenimiento y asegurando una vida útil del sistema más larga.

3.4.2.1 Selección del software para el nodo sensor

Dada la importancia del software en el nodo sensor, su selección se fundamenta en una evaluación exhaustiva de los requerimientos técnicos y operativos del sistema. Es crucial que el entorno de desarrollo elegido sea compatible con la arquitectura de hardware de la placa ESP-32D, ofrezca un soporte robusto para las librerías específicas de comunicación y

sensado, y permita implementar de manera eficiente las funcionalidades previstas de adquisición y preprocesamiento de datos. Las alternativas evaluadas son: Arduino IDE, MicroPython y CircuitPython, en la tabla **Tabla 27** se nota su comparación de acuerdo a los requerimientos establecidos. Esta comparativa permitirá seleccionar la opción óptima para implementar las funciones que ejecutará la placa de manera confiable y eficaz.

Tabla 27

Evaluación de software para nodo sensor

	Requisitos								
	StSR 2	**StSR 3**	**StSR 4**	**SySR 3**	**SySR 9**	**SySR 20**	**SRSH 3**	**SRSH 15**	**Val.**
Arduino IDE	1	1	1	1	1	1	1	1	8
MicroPython	1	1	0	1	1	1	0	1	6
CircuitPython	1	1	0	1	1	1	0	1	6
Cumple “1” No cumple “0”									
Selección: IDE de Arduino									

Nota: *Diseño de autoría propia.*

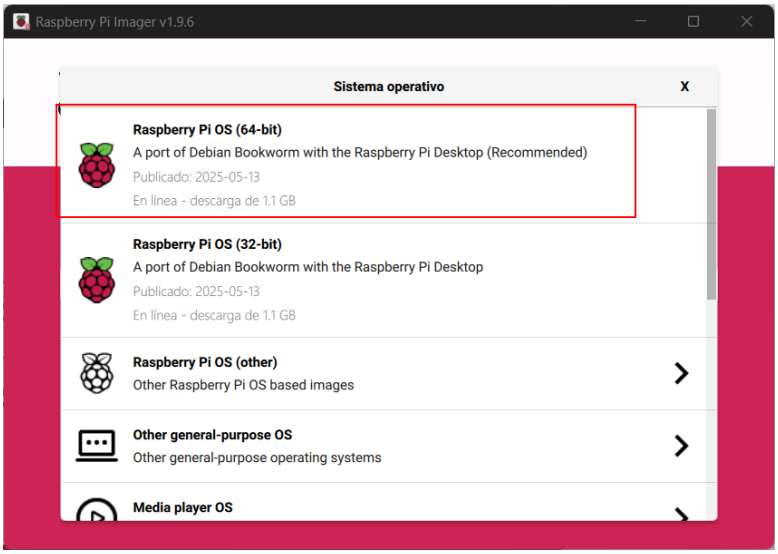
3.4.2.2 Selección del software para el Gateway

Con respecto al Gateway, se seleccionó la solución de software oficial proporcionada por Raspberry Pi: Raspberry Pi OS, un sistema operativo basado en Linux derivado de Debian y optimizado específicamente para su hardware, como se muestra en la **Figura 13**. La decisión de utilizar esta distribución oficial, en lugar de evaluar alternativas genéricas, se fundamenta en que garantiza la máxima compatibilidad, estabilidad y soporte nativo para todos los componentes de la placa, aspectos cruciales para el desempeño confiable y el rol central que

el Gateway cumplirá dentro de la arquitectura de red del sistema FVH.

Figura 13

Sistema Operativo para la placa Rapsberri pi 5 B



Nota: *Diseño de autoría propia.*

Además, esta placa no solo funcionará como Gateway, sino que también se configurará como un servidor unificado. Sobre este sistema operativo se implementarán todas las plataformas esenciales para el proyecto: el almacenamiento de datos, el panel de monitoreo en tiempo real y la lógica de control del sistema. Esta integración centralizada simplifica la arquitectura, reduce la latencia y facilita el mantenimiento, asegurando un funcionamiento robusto y una larga vida útil.

3.4.2.3 Selección de base de datos

La base de datos constituye un pilar fundamental del sistema, ya que sustenta procesos críticos como el entrenamiento del modelo de árbol de decisiones, el monitoreo en tiempo real por parte del usuario final, y la evaluación de métricas clave de eficiencia productiva y verificación del proceso de automatización .Para seleccionar la tecnología más adecuada, se realizó un análisis comparativo de los sistemas de gestión de bases de datos disponibles,

evaluándolos conforme a criterios como el rendimiento en entornos embebidos, la compatibilidad con los lenguajes y plataformas utilizados, la facilidad de administración y su capacidad para operar de manera estable en dispositivos de recursos limitados. Los resultados de esta evaluación se presentan en la **Tabla 28.**

Tabla 28

Evaluación de distintas bases de datos

	Requisitos								
	StSR 2	**StSR 3**	**StSR 4**	**SySR 2**	**SRSH 1**	**SRSH 3**	**SRSH 14**	**SRSH 15**	**Val.**
InfluxDB	1	1	0	1	1	0	1	1	6
PostgreSQL	1	1	0	1	1	0	1	1	6
MySQL	1	1	0	1	1	0	1	1	6
SQLite	1	1	1	1	1	1	1	1	8
MongoDB	1	1	0	1	1	0	1	1	6

Cumple “1” No cumple “0”

Selección: Por consiguiente, la base de datos seleccionada es, **SQlite.**

Nota: *Diseño de autoría propia.*

Tras la evaluación comparativa, todas las opciones de bases de datos presentaron características robustas y una capacidad de respuesta adecuada para el sistema. Sin embargo, el criterio decisivo fue la optimización de recursos, priorizando una solución con bajo consumo de procesamiento y memoria en nuestro servidor y Gateway.

En consecuencia, la base de datos seleccionada es SQLite, misma que se observa en la **Figura 14**. Esta elección se fundamenta en su eficiencia operativa, su fac

ilidad de implementación y su inherente tolerancia a cortes de comunicación. Estas

características se alinean perfectamente con los requisitos de un sistema embebido, garantizando confiabilidad sin comprometer el rendimiento del hardware.

Figura 14

Logo de la base de datos SQLite



Nota: *logotipo obtenido de, (SQLite Home Page, 2025)*

3.4.2.4 Selección de software de monitoreo

Siguiendo la metodología establecida, se procedió a la selección de la plataforma de monitoreo. Esta constituye la interfaz principal del sistema FVH, mediante la cual el usuario final podrá observar, analizar e interactuar con todos los procesos del cultivo hidropónico.

En la selección se priorizo garantizar la seguridad en el tratamiento de la información y optimizar los costos operativos. Por ello, se buscó una solución que ofreciera robustez y confiabilidad sin incurrir en gastos excesivos por servicios o licencias.

Para identificar la opción más adecuada, se realizó una evaluación comparativa de las alternativas disponibles en el mercado. Los resultados de este análisis se presentan en la **Tabla 29**, la cual permitió descartar aquellas opciones que no cumplían con el equilibrio necesario entre usabilidad para el usuario final y la sostenibilidad en costos de implementación y mantenimiento.

Tabla 29

Evaluación para la plataforma de monitoreo

	Requisitos										
	StSR 3	**StSR 5**	**StSR 7**	**StSR 9**	**SySR 1**	**SySR 4**	**SySR 5**	**SRSH 3**	**SRSH 14**	**SRSH 16**	**Val.**
ThingSpeak	1	1	1	1	1	1	1	1	1	1	10
Node-RED	1	1	1	1	1	1	1	1	1	1	10
Azure	1	1	1	0	1	1	1	0	0	1	7
AWS	1	1	1	0	0	1	1	0	0	1	6
Google	1	1	1	0	0	1	1	0	0	1	6

Cumple “1” No cumple “0”

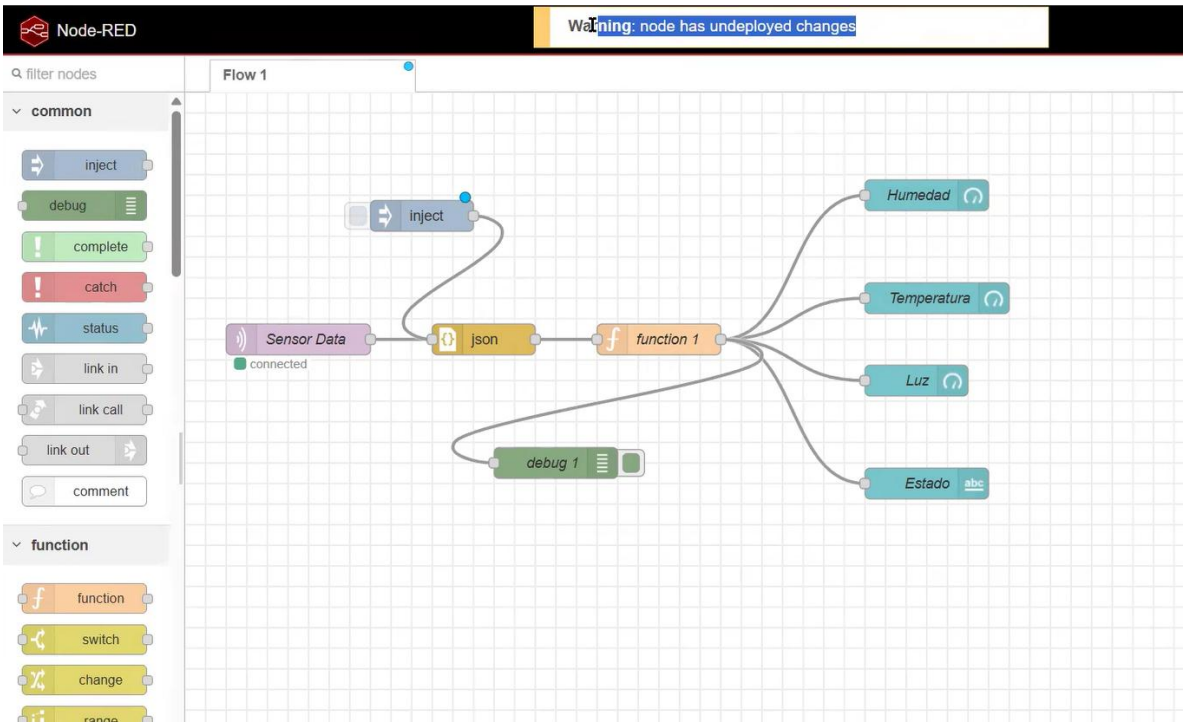
Selección: Por consiguiente, la plataforma seleccionada es **NODE-RED.**

Nota: *Diseño de autoría propia.*

De acuerdo con los resultados de la **Tabla 29**, se determinó que **NODE-RED** es la opción más adecuada para el proyecto, misma que se puede observar en la **Figura 15**. Esta plataforma beneficia al sistema debido a su relativa facilidad de instalación en la placa Raspberry Pi 5B, su naturaleza de código abierto (costo nulo de licencias) y su capacidad para operar de manera offline. Esta última característica es crucial, ya que garantiza la continuidad operativa del sistema incluso ante pérdidas de conectividad a internet, requiriendo únicamente una fuente de energía estable. Además, dentro de la red local, Node-RED puede seguir ejecutando todas sus funciones, incluyendo la generación de alertas y el monitoreo en tiempo real.

Figura 15

Plataforma de monitoreo "Node-Red"



Nota: *se muestra la plataforma node-red implementada en la placa principal.*

3.4.2.5 Software para alertas del sistema

Para la generación de alertas de activación y desactivación del sistema de riego, se seleccionó una plataforma que sea versátil y con una experiencia de usuario intuitiva. La aplicación resultante fue Telegram misma que se muestra en la **Figura 16**, debido a su capacidad para enviar notificaciones inmediatas y de fácil comprensión.

La implementación se configura para que las alertas lleguen al usuario como mensajes directos desde un bot o cuenta dedicada, similar a un chat, al llegar un mensaje de una persona. Este enfoque permite que el usuario monitoree el comportamiento del sistema de forma remota y en tiempo real, recibiendo actualizaciones claras y accesibles sobre el estado del riego.

Figura 16

Logotipo de app Telegram



Nota: *logotipo obtenido de página oficial de la plataforma(Telegram Messenger, n.d.).*

3.4.3 Estructuración del sistema

El proyecto se estructura en dos partes principales que integran los aspectos mecánicos y tecnológicos. El primero corresponde al sistema mecánico, el cual consiste en la infraestructura física para la aspersión y el soporte de las bandejas donde se ubicarán los retoños de trigo, que posteriormente conformarán el Forraje Verde Hidropónico (FVH).

La segunda parte corresponde a la arquitectura IoT, la misma que se encargará de monitorear en tiempo real las condiciones ambientales del cultivo y de controlar de manera autónoma el sistema de riego por aspersión. Esta integración permite la automatización de esta etapa del proceso producir FVH, optimizando los recursos y garantizando las condiciones óptimas para su crecimiento.

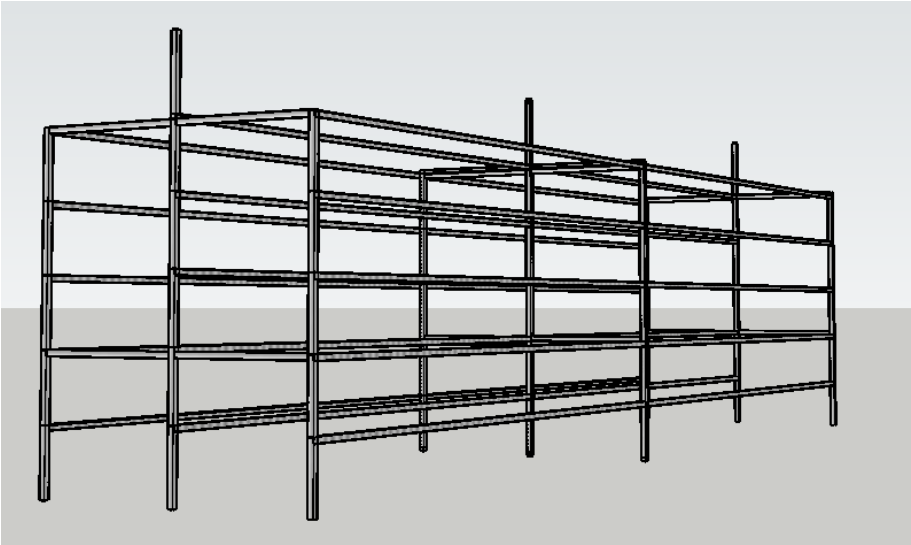
3.4.3.1 Estructuración del riego

Se realizó un diseño preliminar de los estantes donde se ubicarán las bandejas para el desarrollo de las diferentes estepas del FVH. Estas estructuras cuentan con medidas aproximadas de 2.4 m de alto, 2 m de largo y 1.2 m de ancho. Además, se dividen en cuatro sub niveles, en cada uno de los cuales se dispondrán las bandejas de cultivo.

Una referencia de estos es cómo podemos notar en la **Figura 17**, ya que este fue diseñado en un programa de modelado 3D este nos servirá como referencia para la instalación real del sistema.

Figura 17

Estantes metálicos para FVH



Nota: *Estantes para FVH diseñados en SKETCHUP.*

3.4.3.1.1 Sistema de aspersión

Este sistema es uno de los más utilizados en la producción de FVH. Su configuración incluye una bomba de agua que genera la presión necesaria para impulsar el agua a través de una red de mangueras guía, ubicadas estratégicamente en cada sección de los estantes.

Estas mangueras cuentan con microaspersores integrados, los cuales distribuyen el agua de manera uniforme sobre las bandejas de cultivo, logrando así un eficiente riego por aspersión en todas las etapas de crecimiento del forraje.

Figura 18

Microaspersores que serán utilizados para el riego



Nota: *Imagen de autoría propia, microaspersores a utilizarse.*

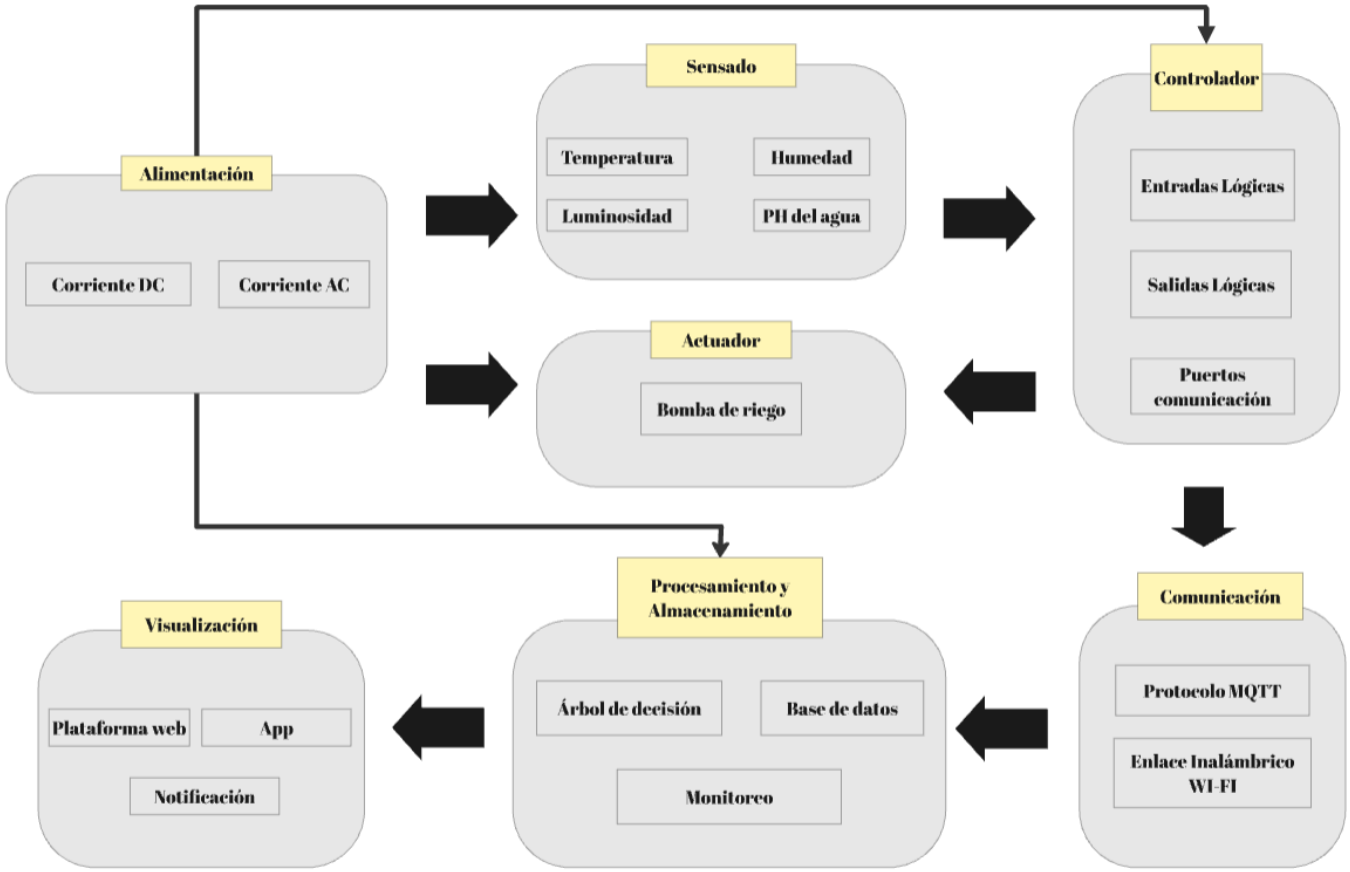
3.4.3.3 Diagrama de bloques del sistema

El diagrama de bloques del sistema, representado en la **Figura 19**, está conformado por siete bloques interconectados que ilustran el flujo de relación y las funciones entre los componentes. La secuencia operativa inicia con el bloque de Alimentación, continúa con el de Sensado de Datos que este paralelo con el Control de Actuadores.

La información es gestionada por el Controlador, el bloque de Comunicación permite que se envié los datos al bloque de Procesamiento y Almacenamiento para culminar en la interfaz de Visualización. A continuación, se describe la función específica de cada uno de estos bloques.

Figura 19

Diagrama de Bloques del sistema



Nota: *Diseño de autoría propia.*

A. Alimentación

El bloque de alimentación constituye la base energética del sistema de riego automático y garantiza un funcionamiento estable y seguro de todos los componentes electrónicos. Este módulo incluye dos tipos principales de suministro: corriente continua (DC), utilizada para alimentar el microcontrolador, los sensores, la placa de procesamiento, y corriente alterna (AC), destinada principalmente a la alimentación de la bomba de riego. La adecuada gestión energética en este bloque asegura que los cambios bruscos de corriente o ruido eléctrico no afecten la lectura de los sensores ni la estabilidad de la comunicación con la Raspberry Pi 5, manteniendo así la integridad del sistema en condiciones operativas exigentes.

B. Sensado

El bloque de sensado es el responsable de adquirir en tiempo real las variables ambientales que determinan el estado del entorno y permiten automatizar el proceso de riego. Este módulo integra sensores especializados para medir temperatura, humedad, luminosidad y pH del agua, utilizando protocolos de comunicación específicos para cada dispositivo (I^2C, señal analógica y comunicación single-wire).

El sensor capacitivo de humedad del suelo proporciona una medición estable del contenido hídrico sin degradación por corrosión; el sensor BH1750 ofrece valores precisos de iluminancia en lux mediante comunicación digital I^2C; el sensor DHT22 obtiene datos de temperatura y humedad ambiental; y el sensor de pH monitorea la acidez del agua para garantizar que se mantenga en un estado óptimo para el cultivo. Los datos recogidos por este bloque son muestreados periódicamente y transmitidos al microcontrolador ESP32.

Este bloque es esencial para el sistema, ya que suministra la información base sobre la cual se aplican los algoritmos de decisión y análisis que regulan el funcionamiento automatizado del sistema FVH.

C. Actuador

El bloque de actuadores tiene la función de transformar las señales digitales generadas por el controlador en acciones físicas, principalmente la activación o desactivación de la bomba de riego. Para ello, se utilizan módulos de potencia como son los relés y contactores capaces de manejar los requisitos eléctricos de la bomba ya sea en corriente continua o alterna. Este bloque incluye las protecciones necesarias para garantizar operaciones seguras, como diodos flyback, fusibles y aislamiento eléctrico. Su diseño asegura que la bomba responda de manera confiable a las órdenes enviadas por el ESP32, minimizando riesgos de fallos y garantizando la disponibilidad del sistema. El correcto funcionamiento del actuador es crítico, ya que constituye el componente encargado de ejecutar el riego de forma automática o supervisada.

D. Controlador (ESP32)

El controlador, basado en el microcontrolador ESP32, es el encargado de gestionar la lectura de sensores, controlar los actuadores y establecer comunicación con la Raspberry Pi 5. Este dispositivo cuenta con entradas analógicas para captar señales provenientes del sensor de humedad del suelo, así como con entradas digitales para sensores ambientales y con salidas digitales para el manejo del relé que activa la bomba.

El ESP32 ejecuta procesos de preprocesamiento de datos, filtrado de ruido, verificación de umbrales, y toma de decisiones básicas en caso de fallos de comunicación. Su capacidad de cómputo y su arquitectura de bajo consumo lo

convierten en un componente ideal para el control de sistemas IoT distribuidos y orientados al tiempo real.

E. Comunicación

El bloque de comunicación define el mecanismo que permite la interacción entre el microcontrolador ESP32 y la Raspberry Pi 5, así como su posible conexión con servicios externos. Este Bloque se basa en el protocolo MQTT, el cual opera mediante un esquema de publicación y suscripción que optimiza el uso del ancho de banda y permite un envío eficiente y confiable de mensajes, incluso en redes de baja estabilidad. El ESP32 actúa como publicador de datos de los sensores y como suscriptor de comandos generados por el sistema de procesamiento. La comunicación se efectúa a través del enlace inalámbrico WLAN en base al estándar 802.11ac, lo que permite integrar el sistema en una red local doméstica o institucional y habilitar su acceso remoto. Este bloque asegura la sincronización continua entre el controlador y la Raspberry Pi, manteniendo el flujo de datos necesario para la operación autónoma del sistema de riego.

F. Procesamiento y Almacenamiento (Raspberry Pi 5)

La Raspberry Pi 5 actúa como servidor de análisis y procesamiento avanzado, ejecutando algoritmos de toma de decisiones que requieren una capacidad de cómputo mayor que la disponible en el ESP32. En este módulo se implementa un modelo de árbol de decisión, entrenado previamente con datos históricos para identificar patrones en la humedad del suelo, la temperatura, la luminosidad y el comportamiento del sistema. Este modelo determina el momento óptimo de activación del riego, mejorando la eficiencia del uso del agua. Además, la Raspberry Pi gestiona una base de datos encargada de almacenar mediciones, eventos del

sistema y registros de activación de la bomba, permitiendo análisis históricos y auditoría del funcionamiento. Las funciones de monitoreo supervisan el estado del controlador, los sensores y la red, generando alertas en caso de fallos o condiciones anómalas. Este bloque convierte el sistema en una plataforma inteligente de automatización agrícola.

G. Visualización

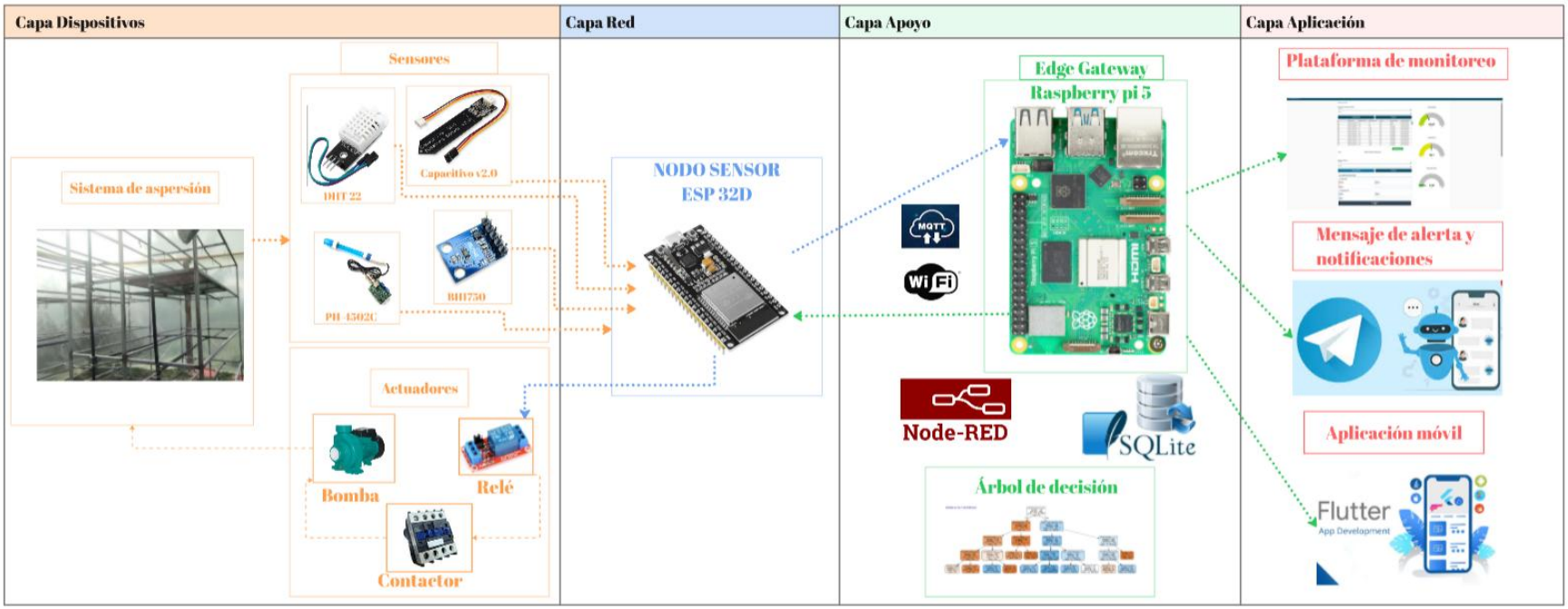
El bloque de visualización proporciona al usuario final una interfaz accesible para supervisar y gestionar el sistema de riego automático. A través de una plataforma web alojada en la Raspberry Pi y de una aplicación móvil, el usuario puede revisar en tiempo real las variables captadas por los sensores, los estados de la bomba, los registros históricos y las decisiones tomadas por el algoritmo. La disponibilidad de esta interfaz asegura una interacción intuitiva que facilita la supervisión remota y el manejo del sistema desde cualquier dispositivo conectado a la red.

3.4.3.3 Arquitectura general

Tras la explicación del funcionamiento del sistema, se presenta la arquitectura general para facilitar la comprensión de la composición del sistema de automatización IoT. Esta arquitectura, ilustrada en la **Figura 20**, integra todos los dispositivos previamente seleccionados en etapas anteriores, los cuales fueron elegidos por ser los más adecuados para los requisitos específicos del proyecto.

Figura 20

Arquitectura general del sistema de FVH



Nota: *Diseño de autoría propia.*

3.5 Calculo de consumo de corriente del sistema.

Un aspecto crucial en el diseño del sistema es el cálculo del consumo de corriente, determinado por el voltaje de operación y las características de funcionamiento de cada componente. Este análisis incluye tanto los circuitos electrónicos de baja potencia (sensado y control) como los circuitos de potencia que gobiernan la bomba de riego mediante un contactor. Este actuador funciona como un enclave de activación que suministra energía de manera segura y controlada a la bomba, permitiendo así la ejecución automatizada del riego en el proceso de desarrollo del Forraje Verde Hidropónico (FVH).

En la **Tabla 30** se desglosa el voltaje operativo y el consumo de corriente de cada uno de los componentes que integran el sistema automatizado, distinguiendo entre el subsistema electrónico (sensores, microcontroladores, comunicaciones) y el subsistema eléctrico (bomba, contactor, fuentes de alimentación). Adicionalmente, se presenta el consumo total estimado del sistema, información crucial para el dimensionamiento correcto de la fuente de alimentación y la evaluación de la eficiencia energética global del proyecto.

Tabla 30

Consumo de corriente en DC de los componentes del sistema.

Nº	Dispositivo	Corriente Activa (mA)	Corriente Sleep (mA)	Voltaje
1	**ESP-WROOM-32D**	160	10	3.3V DC
2	**Raspberry Pi 5B**	5000	500	5V DC
3	**DHT22**	1.5	0.05	3.3–5V DC
4	**Capacitive V2.0**	10	5	3.3–5.5V DC
5	**BH1750**	0.12	0.01	2.4–3.6V DC

6	**PH-4502C**	10	5	5.2V DC
Consumo total DC activo: 5181.62 mA				
Consumo total DC en sleep: 515.06 mA				

Nota: *Diseño de autoría propia.*

En la **Tabla 31** se presenta el consumo de los actuadores que operan con corriente alterna (AC), detallando los valores durante su estado activo:

Tabla 31

Consumo de actuadores en AC

Dispositivo	**Corriente (A)**	**Potencia (W)**	**Consumo en 5 min (Wh)**
Contactor	0.636 A	69.96 W	5.81 Wh
Bomba Paolo	11 A	1210 W	100.43 Wh

Nota: *Diseño de autoría propia.*

El análisis energético del sistema IoT se realizó considerando los estados de operación activa y modo de bajo consumo (sleep), enfatizando que durante la noche las variaciones de temperatura y humedad son menos influyentes, permitiendo ciclos de suspensión intercalados con breves periodos activos. Específicamente, desde las 18:00 horas, el sistema entra en modo sleep durante 3 horas y se activa 1 hora, repitiendo este ciclo hasta las 09:00 horas, cuando permanece activo continuamente hasta las 17:59 horas.

El consumo promedio ponderado del sistema se determinó mediante la ecuación:

$$\boldsymbol{I_{prom}} = \frac{(\boldsymbol{T_{ma}} * \boldsymbol{I_{ma}}) + (\boldsymbol{T_{ms}} * \boldsymbol{I_{ms}})}{(\boldsymbol{T_{ma}} + \boldsymbol{T_{ms}})} \quad (1)$$

Donde

- $\boldsymbol{T_{ma}}$ = Tiempo en modo activo
- $\boldsymbol{I_{ma}}$ = Corriente en modo activo
- $\boldsymbol{T_{ms}}$ = Tiempo en modo sleep
- $\boldsymbol{I_{ms}}$ = Corriente en modo sleep
- $\boldsymbol{I_{prom}}$ = Corriente promedio

Considerando el análisis previo, se establece que el sistema opera durante 12 horas en modo activo y 12 horas en modo de bajo consumo (sleep) a lo largo del día. Aplicando estos valores a la ecuación de corriente promedio ponderada, se obtiene el consumo representativo del sistema en un ciclo completo de 24 horas.

Donde:

$\boldsymbol{T_{ma}}$ = 12 horas

$\boldsymbol{I_{ma}}$ = 5181.62 mA

$\boldsymbol{T_{ms}}$ = 12 horas

$\boldsymbol{I_{ms}}$ = 515.06 mA

Sustituyendo los datos en la ecuación:

$$\boldsymbol{I_{prom}} = \frac{(12 * 5181.62\ ) + (12 * 515.06)}{(12 + 12)} \quad \ldots(2)$$

$$\boldsymbol{I_{prom}} = \frac{(62179.44) + (6180.72)}{(24)} \quad \ldots(3)$$

$$\boldsymbol{I_{prom}} = \frac{68360.16}{24} \quad \ldots(4)$$

$$\boldsymbol{I_{prom}} = \ \mathbf{2848.34\ mA} \quad \ldots(5)$$

El resultado indica que el consumo promedio del sistema en un día típico es de 2848.34 mA, valor que permite dimensionar adecuadamente la fuente de alimentación y estimar la autonomía requerida para garantizar la operación continua del sistema FVH.

3.6 Lógica del sistema

En el presente apartado se analiza la lógica de funcionamiento de cada componente del sistema, correspondiente tanto al microcontrolador como a la placa Gateway/servidor. Para ello, se emplean diagramas de flujo que modelan los procesos y la toma de decisiones de cada subsistema. Consecuentemente, se presentan los diagramas específicos para cada sensor, así como para las dos placas principales.

3.6.1 Estructuración del sistema IoT

Esta sección se explica el funcionamiento de todos los componentes electrónicos del sistema IoT. El análisis comienza con una descripción detallada de los diagramas, que incluyen el diagrama de bloques general para entender el comportamiento global y la arquitectura del sistema que interrelaciona todos sus componentes.

Se continua, se detallando los diagramas de flujo de cada subsistema, explicando su lógica de operación y secuencia de procesos. Para finalizar, se documenta la fase de pruebas y calibración de los sensores, así como la programación e implementación del software tanto en el nodo como del gateway, garantizando así la comunicación y el funcionamiento integral de todo el sistema.

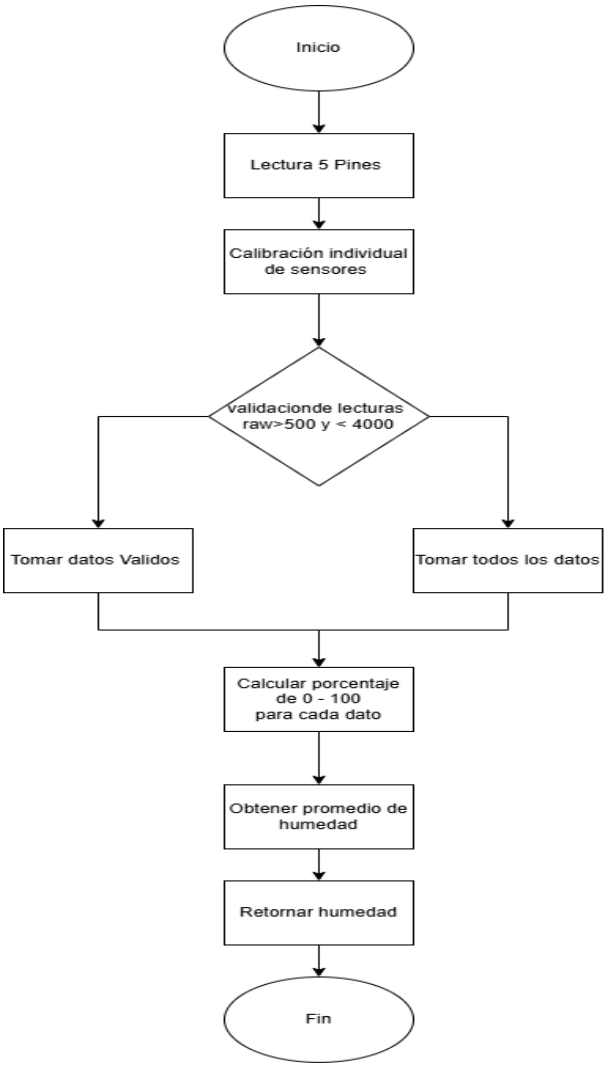
3.6.1.1 Diagrama de flujo del sensor de humedad

Como se observa en la **Figura 21**, el proceso de sensado de humedad se inicia con la lectura de la señal proveniente de los sensores capacitivos versión 2.0 instalados en diferentes puntos del sistema. Los datos obtenidos son enviados a la placa ESP32, donde se ejecuta un algoritmo de calibración que ajusta las lecturas para mejorar su precisión. Posteriormente, las muestras ingresan a un bucle de evaluación que, valida la consistencia de los datos,

descartando aquellas lecturas que puedan corresponder a fallos del sensor o interferencias. Este filtro evita la generación de falsos positivos o negativos en la detección de humedad. Finalmente, los valores validados son convertidos a un rango porcentual de 0% a 100% y transmitidos a la placa principal para su integración en la lógica de control del sistema de riego.

Figura 21

Diagrama de flujo del sensado de humedad



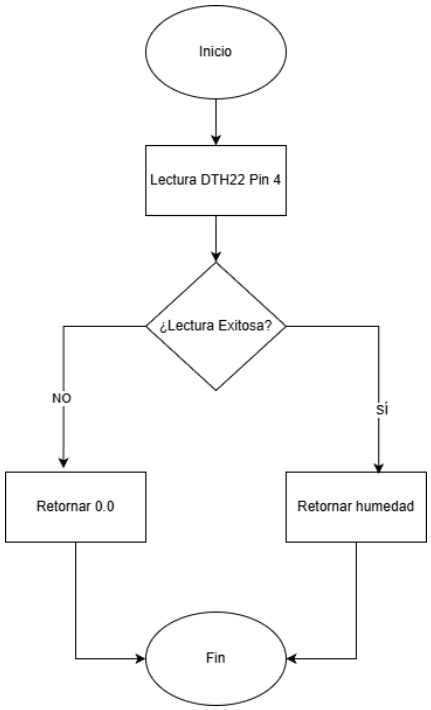
Nota: *Diseño de autoría propia.*

3.6.1.2 Diagrama de flujo del sensor de temperatura

Ya que este tipo de sensores cuentan con su propio protocoló de comunicación single-wire y además mantiene su propia librería que establecen el tipo de lecturas el dato de este sensor es más preciso y con una tolerancia menor al -+0.5%. Este sensor proporciona mediciones digitales de humedad relativa y temperatura más confiables, el sistema verifica si la lectura se ha completado exitosamente. En caso de que la comunicación con el sensor falle o se detecten datos corruptos, el procedimiento retorna un valor de 0.0% como código de error. Cuando la lectura es válida, el sistema retorna el valor real de humedad relativa medido por el sensor, expresado en porcentaje (0-100%). Finalmente, el proceso concluye habiendo proporcionado ya sea un dato confiable que se enviara a la placa principal, esto podemos observar en la **Figura 22**.

Figura 22

Diagrama de flujo del sensado de temperatura



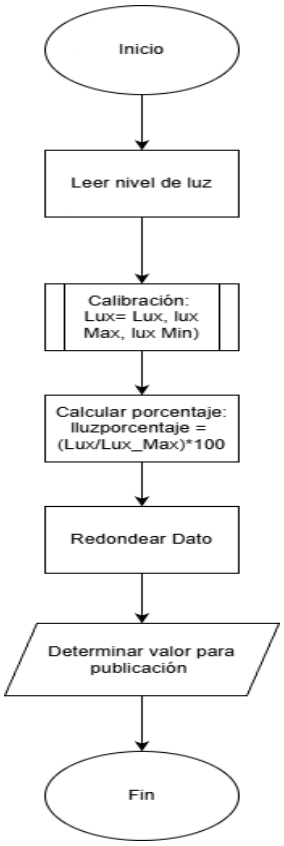
Nota: *Diseño de autoría propia.*

3.6.1.3 Diagrama de flujo de sensor de luz

De acuerdo con las características del sensor, se ha diseñado el diagrama de flujo que define su funcionamiento como se muestra en la **Figura 23**. Cabe destacar que este dispositivo utiliza el protocolo de comunicación I2C para transmitir los datos de luminosidad (lux) al sistema. Durante la ejecución del bucle principal del ESP32, se leen los datos proporcionados por el sensor, los cuales se procesan para establecer el valor de la intensidad lumínica capturada. Posteriormente, se calcula el porcentaje de luz equivalente a la lectura obtenida, y este valor se retorna para ser enviado y publicado en el servidor MQTT.

Figura 23

Diagrama de flujo del sensado de luz



Nota: *Diseño de autoría propia.*

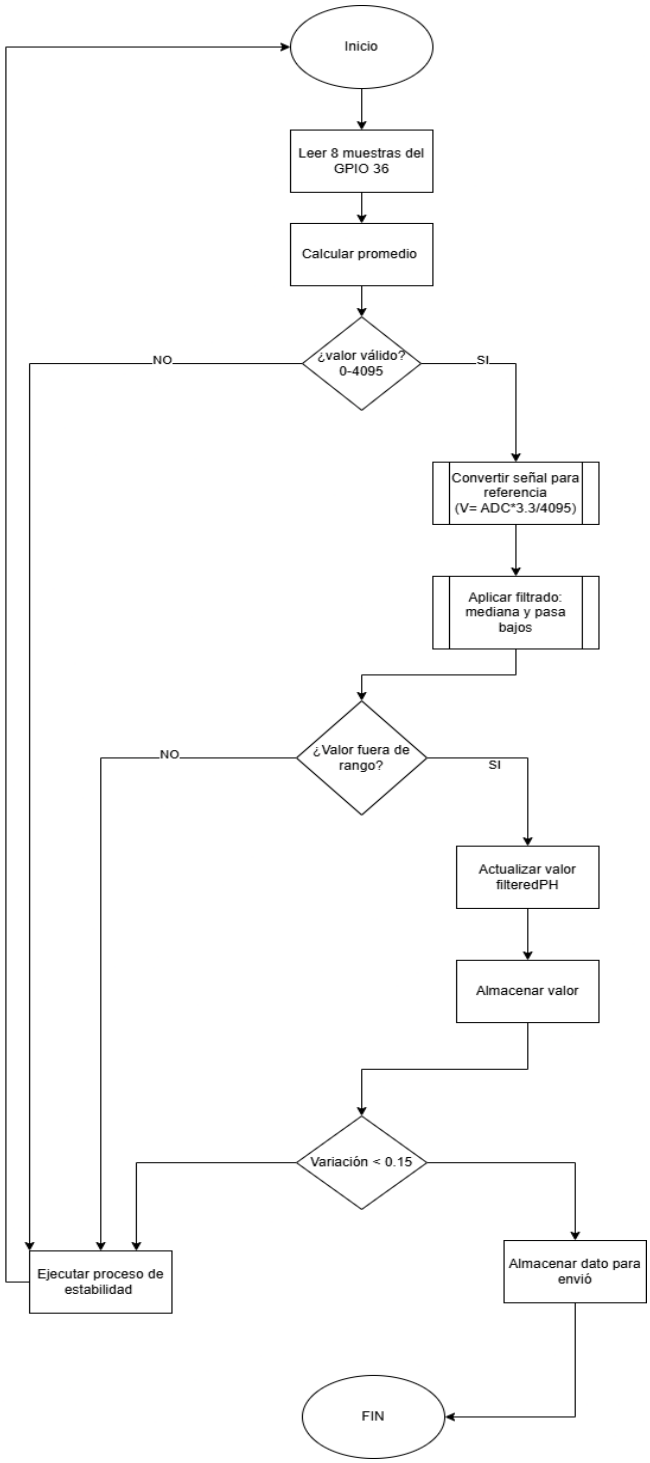
3.6.1.4 Diagrama de flujo del sensor de pH

El diagrama de flujo mostrado en la **Figura 24,** correspondiente al sensor de pH PH-4502C, presentado en la figura correspondiente, contempla un proceso inicial de estabilización de la lectura, necesario debido a la naturaleza electroquímica del sensor. Posteriormente, se implementa la toma de múltiples muestras en un intervalo definido para calcular un valor promedio, minimizando así el impacto de lecturas erróneas o ruido en la señal.

Como parte del procesamiento, se aplican filtros y métodos de descarte basados en desviación estándar, eliminando valores atípicos que puedan distorsionar la medición. Finalmente, el sistema evalúa las lecturas válidas y almacena el dato procesado para su próximo envío al Gateway, asegurando de esta manera que la información transmitida sea confiable, precisa y representativa del estado real del pH.

Figura 24

Diagrama de flujo de lectura del sensor de pH



Nota: *Diseño de autoría propia.*

3.6.1.5 Diagrama de flujo general del nodo

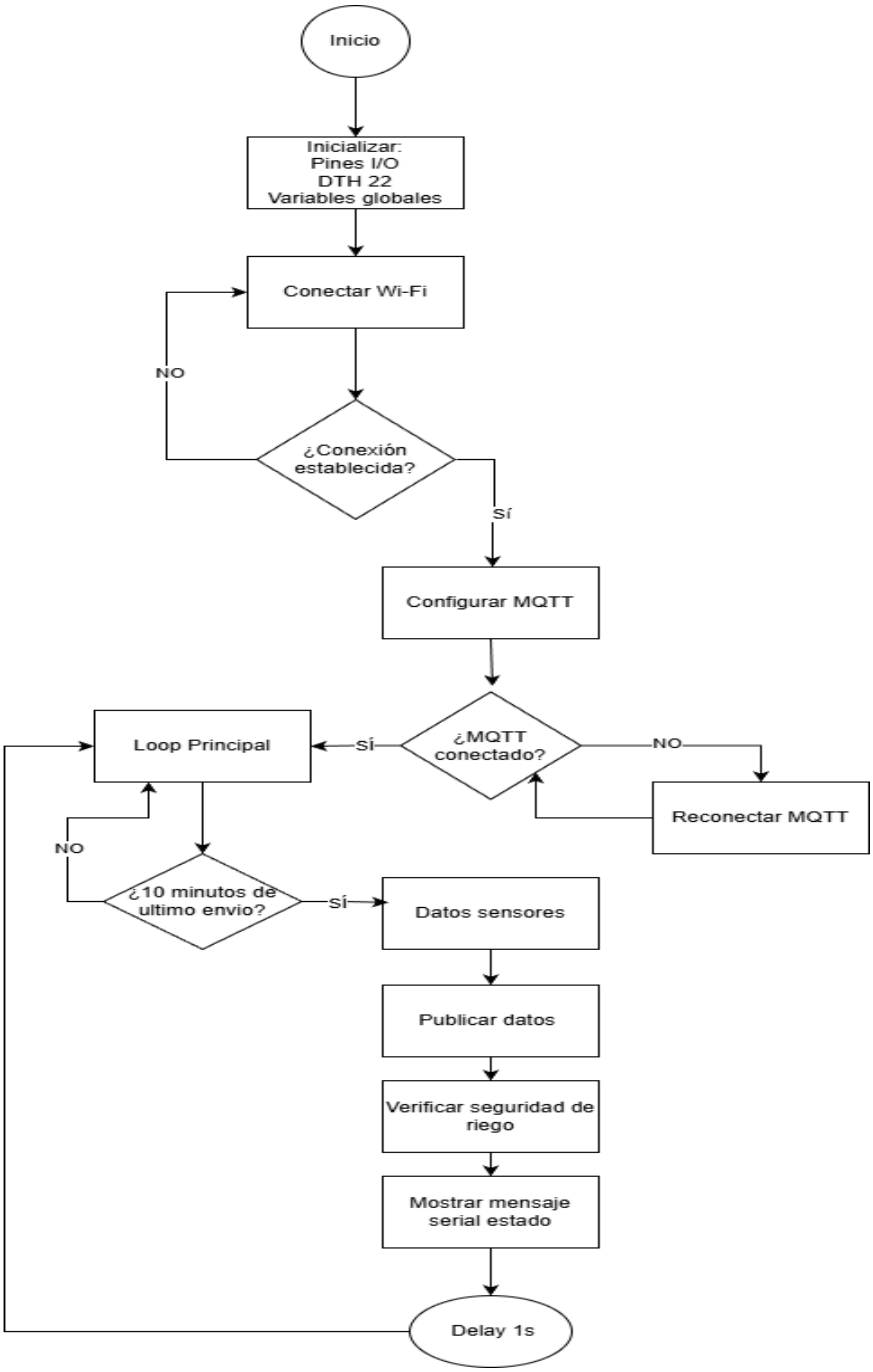
Una vez descritos los flujos de los sensores que serán utilizados en el sistema IoT, se procede a detallar el funcionamiento de la placa ESP32, que actúa como nodo del sistema. Al iniciar el programa, se inicializan los pines de entrada y salida, se declaran las variables globales y se incluyen todas las librerías necesarias para su operación. Una vez completada esta configuración, se establece el enlace inalámbrico mediante el estándar 802.11 (Wi-Fi). Tras lograr la conexión, se configura el protocolo de comunicación MQTT, verificando su correcta inicialización; caso contrario, se reintenta hasta lograr establecerla exitosamente.

Confirmada la comunicación, se ejecuta el bucle principal, donde se realizan las tareas de sensado y preparación de datos. Transcurrido el periodo de espera establecido, los datos de los sensores se publican a la placa principal y se recibe la decisión de activar o desactivar el riego. En caso de activación, el sistema evalúa las condiciones de seguridad, desactivando automáticamente el riego después de 5 minutos. Finalmente, se imprime el estado del sistema por el puerto serial y se introduce un delay antes de repetir el ciclo.

Todo lo expuesto podemos notar en la **Figura 25**.

Figura 25

Diagrama de flujo del nodo sensor



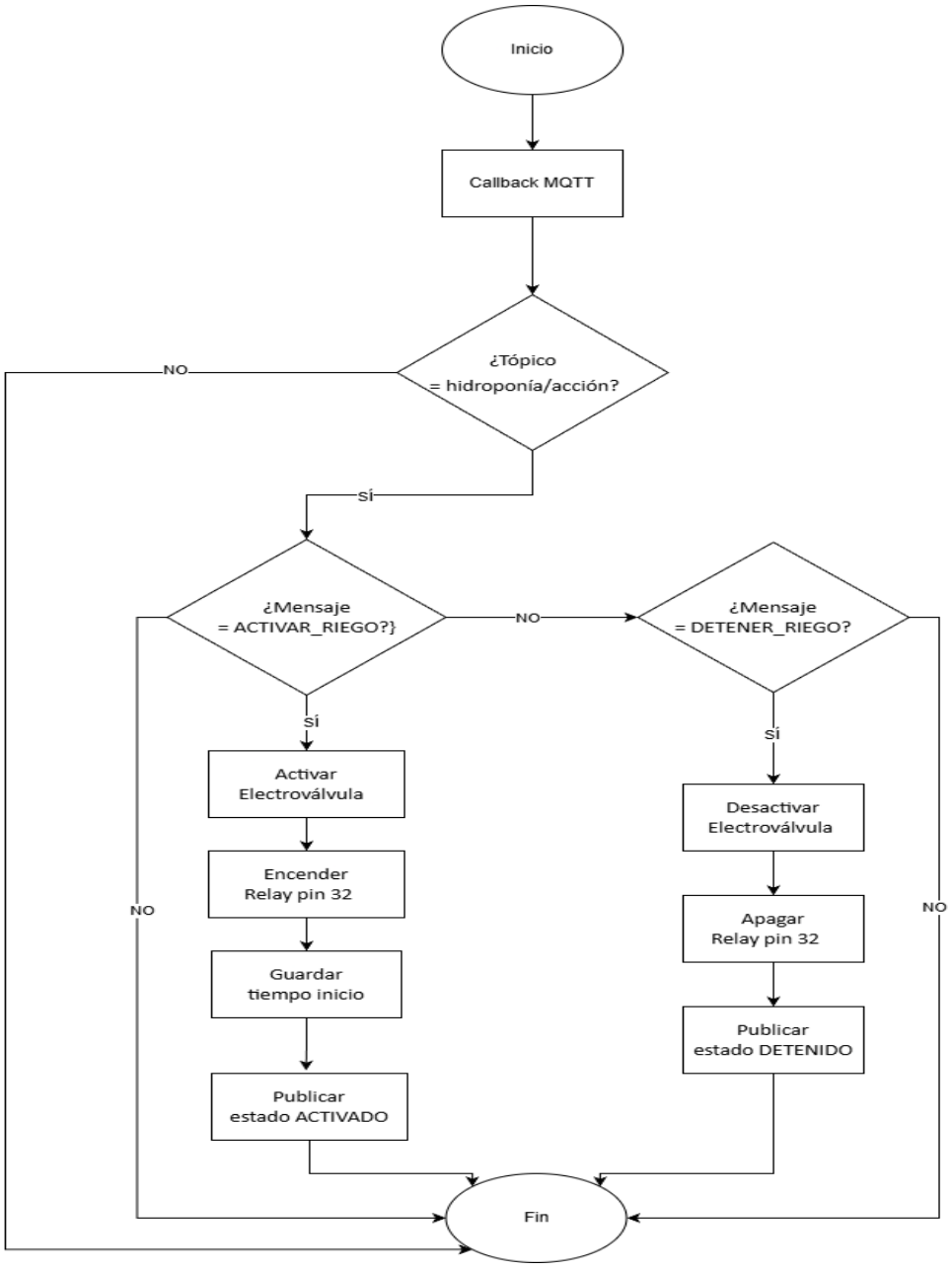
Nota: *Diseño de autoría propia.*

3.6.1.6 Control de Riego mediante Protocolo MQTT

El control del sistema de riego se gestiona desde la placa principal a través de mensajes callback enviados mediante el protocolo MQTT. Cuando se recibe un mensaje en el tópico de suscripción hidroponia/action, el sistema evalúa su contenido para determinar la acción correspondiente: activar o desactivar el riego. Una vez validada la instrucción, se envía la orden al pin designado del microcontrolador, el cual acciona o desactiva el relé que gobierna la bomba. Este flujo de control se ilustra en el diagrama presentado en la **Figura 26**, donde se detalla el proceso de comunicación y ejecución de la orden de riego.

Figura 26

Diagrama de control del riego



Nota: *Diseño de autoría propia.*

3.6.1.7 Diagrama del Gateway servidor

Para la parte de procesamiento que es la placa principal. El flujo inicia con la puesta en marcha del sistema, es decir se ejecuta los archivos de configuración donde se cargan los modelos de Machine Learning, se verifica o crea la base de datos local y se establece la

conexión con el broker MQTT. después, el gateway entra en un estado de escucha continua a la espera de mensajes provenientes del microcontrolador a través del protocolo MQTT.

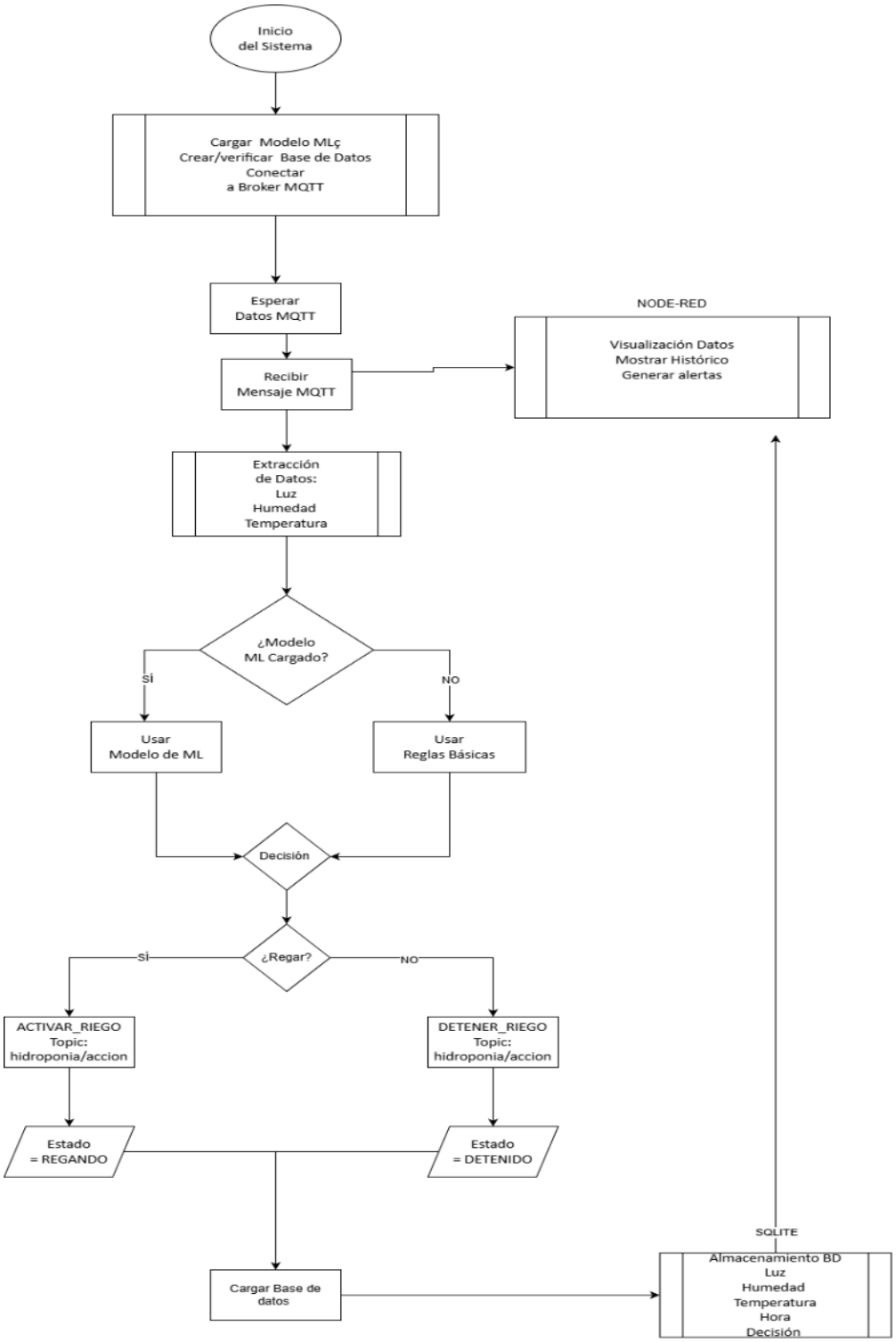
Cuando se recibe un mensaje, el sistema procede a la extracción y separación de los datos ambientales, como son los datos de luz, humedad de suelo y temperatura. Posteriormente, el flujo verifica si el modelo de Machine Learning ha sido cargado correctamente. En caso afirmativo, la decisión sobre activar o no el riego se realiza mediante dicho modelo; en caso contrario, se aplican reglas básicas predefinidas en el sistema.

Después pasamos a la fase de decisión donde se es determina si es necesario regar. Si la decisión es afirmativa, el gateway publica el comando ACTIVAR_RIEGO en el tópico correspondiente del broker MQTT y actualiza el estado del sistema a “REGANDO”. Si la decisión es negativa, se emite el comando DETENER_RIEGO y el estado pasa a “DETENIDO”. Una vez ejecutada la acción, los datos recibidos incluyendo luz, humedad, temperatura, fecha, hora y decisión tomada se almacenan en la base de datos SQLite para el registro histórico.

Paralelamente, el sistema dispone de un módulo destinado a visualización, el cual permite mostrar la tabla de registros, monitorear mediante gráficas y generar alertas basadas en la información almacenada. De esta manera se muestra en la **Figura 27**, el diagrama que describe el funcionamiento integral del Gateway.

Figura 27

Diagrama de Flujo de la placa central



Nota: *Diseño de autoría propia.*

3.7 Configuraciones del sistema

En el siguiente apartado se enfoca en describir el código que se implementara, basado en los diagramas de flujo previamente definidos, con el objetivo de garantizar el funcionamiento esperado del sistema de FVH. Se detallan los métodos de programación, librerías y lógica utilizados tanto en el nodo principal (ESP32) como en la placa servidor (Raspberry Pi), encargados de la adquisición de datos, el procesamiento y el monitoreo en tiempo real del sistema. Adicionalmente, se incluyen los diagramas de conexión electrónica para cada componente del sistema, lo que permitirá replicar físicamente las instalaciones mediante una guía de cableado e interconexión de sensores y actuadores.

3.7.1 Configuración sensores

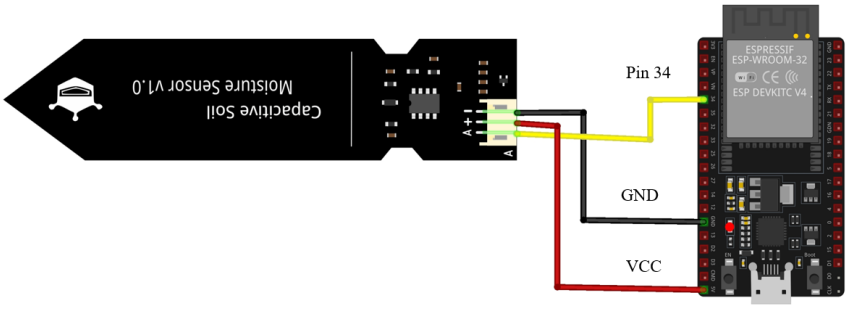
Inicialmente, se simuló el circuito de cada sensor mediante el software Fritzing para validar las conexiones eléctricas y su funcionamiento. Posteriormente, se desarrolló el código específico para cada sensor de manera individual, implementando algoritmos de calibración que permitieron ajustar las lecturas, optimizar su precisión y garantizar la confiabilidad de los datos capturados por el sistema.

3.7.1.1 Configuración del sensor de humedad

Como se observa en la **Figura 28**, el sensor de humedad capacitivo versión 2.0 (HW-390) presenta tres conexiones básicas: alimentación (VCC), tierra (GND) y señal. Durante la fase de pruebas, el sensor se conecta a la placa ESP32 utilizando los pines de alimentación y tierra correspondientes, mientras que el pin de señal se conecta específicamente al GPIO 34 del nodo para la lectura de datos.

Figura 28

Conexiones del sensor de humedad



Nota: *Diseño de autoría propia.*

Una vez establecidas las conexiones del sensor de humedad, se procedió a desarrollar el código de calibración. Dado que este sensor proporciona una señal analógica, se utilizó la función analogRead para capturar las lecturas. El proceso de calibración consistió en comparar los valores obtenidos en dos estados extremos: con el sensor completamente sumergido en agua y en condiciones de sequedad total. Con estos datos de referencia, se implementó la **Ecuación 6** para convertir las lecturas en porcentaje de humedad:

$$\boldsymbol{H}(\%) = 100.\frac{(\mathbf{Vs} - \mathbf{Vr})}{(\mathbf{Vs} - \mathbf{Vm})} \quad \ldots(6)$$

Donde:

H, es el porcentaje de humedad del suelo

Vs, es el valor de lectura del sensor en suelo seco.

Vm, es el valor de lectura del sensor al estar en agua completamente.

Vr, es el valor de lectura del sensor al momento de la medición.

Finalmente, mediante la función constrain se delimitaron los valores de salida al rango de 0% a 100%, asegurando que el porcentaje de humedad calculado se mantenga dentro de los límites esperados. Esta implementación puede verificarse en el código de la **Figura 29**, donde se observa la declaración de variables y la aplicación de la función de calibración.

Figura 29

Código para el sensor de Humedad

```
const int pinHumedad = 34; // AO del módulo al GPIO34 del ESP32

// Valores de calibración (ajústalos según tus pruebas)
const int VALOR_SECO = 3200;   // Valor cuando el sensor está completamente seco (en aire)
const int VALOR_MOJADO = 1250;  // Valor cuando el sensor está completamente sumergido en agua

float leerHumedadSuelo() {
  int soilValue = analogRead(pinHumedadSuelo);
  // Mapear correctamente los valores del sensor de humedad
  float soilPercentage = 100.0 * (VALOR_SECO - soilValue) / (VALOR_SECO - VALOR_MOJADO);
  soilPercentage = constrain(soilPercentage, 0, 100);
  return soilPercentage;
}
```

Nota: *Diseño de autoría propia.*

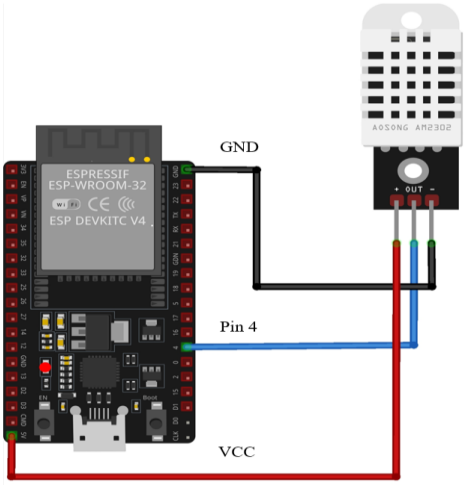
Con el sensor ya configurado se puede poner a prueba de las lecturas y porcentaje medido por el momento y a la vez se puede implementar en el código principal.

3.7.1.2 Configuración del sensor de temperatura

De igual forma, se simuló el circuito del sensor de temperatura en la plataforma Fritzing, sabiendo que la placa cuenta con pines de VCC y GND para la alimentación del sensor. Para la comunicación de datos, se asignó el pin GPIO 4, como se evidencia en la **Figura 30**.

Figura 30

Conexiones del sensor DTH22 con la placa ESP 32D



Nota: *Diseño de autoría propia.*

Este sensor utiliza la librería específica proporcionada por el entorno Arduino, compatible con su protocolo de comunicación "single-wire". Para su implementación, se importó dicha librería y se configuró el pin GPIO 4 como interfaz de comunicación tal como vemos en la **Figura 31**. Dado su reducido margen de error (±5%), las lecturas obtenidas se validaron mediante comparación con fuentes confiables de temperatura, permitiendo verificar la precisión del sensor y determinar posibles variaciones en las mediciones.

Figura 31

Código para el sensor DTH22

```
4   #include <DHT.h>
21  #define DHTPIN 4
22  #define DHTTYPE DHT22
131 float leerTemperatura() {
132   float temp = dht.readTemperature();
133   if (isnan(temp)) {
134     Serial.println("Error leyendo temperatura");
135     return 0.0;
136   }
137   return temp;
138 }
```

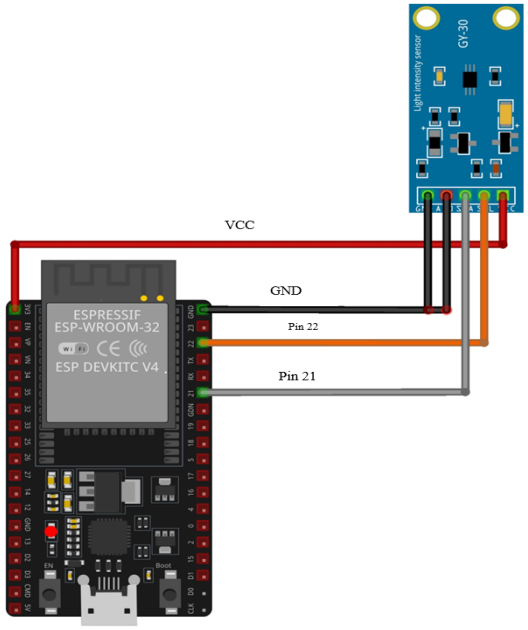
Nota: *Diseño de autoría propia.*

3.7.1.3 Configuración del sensor de luz

El sensor de luz BH1750 se conectó al nodo ESP32 utilizando los pines de alimentación correspondientes (GND y VCC). Debido a que este sensor emplea el protocolo de comunicación I2C, se establecieron las conexiones de los pines SDA (datos) y SCL (reloj) con los pines equivalentes del microcontrolador. Esta configuración, tal como se evidencia en la **Figura 32**, garantiza una comunicación adecuada y sincronizada para la adquisición precisa de los niveles de luminosidad en el entorno de cultivo.

Figura 32

Conexiones para el sensor BH1750 con la placa ESP 32D



Nota: *Diseño de autoría propia.*

Para establecer el porcentaje de luminosidad a partir de las lecturas en lux, se definieron los valores de referencia en condiciones extremas: LUX_MIN (oscuridad total) y LUX_MAX (iluminación máxima). La normalización se realizó mediante la función constrain(lux, LUX_MIN, LUX_MAX) para limitar el rango de medición, aplicando posteriormente la

ecuación de porcentaje:

$$\boldsymbol{LUZ}(\%) = 100.\left(\frac{lux}{(LUX_MAX)}\right) \# \ldots (7)$$

Donde:

LUZ, es el porcentaje de humedad de

lux, es el valor de lectura del sensor.

LUX_MAX, es el valor de lectura del sensor en máxima exposición de luz.

LUX_MIN, es el valor de lectura del sensor en mínima exposición de luz.

Como se muestra en la **Figura 33**, el código implementado utiliza la librería específica del sensor, definiendo los pines de comunicación I²C e inicializando el protocolo serial. En el bloque de ejecución principal se realiza la lectura del sensor, se aplica la restricción de rango y se calcula el porcentaje de luminosidad para su envío.

Figura 33

Código para el Sensor de luz

```
3   #include <BH1750.h>
8   const float LUX_MIN = 0.0;      // Oscuridad total
9   const float LUX_MAX = 1500.0;   // Bien iluminado interior
149 float leerLuz() {
150   float lux = sensorLuz.readLightLevel();
151
152   // Calcular porcentaje con límite práctico
153   lux = constrain(lux, LUX_MIN, LUX_MAX);
154   float lightPercentage = (lux / LUX_MAX) * 100.0;
155   lightPercentage = constrain(lightPercentage, 0, 100);
156
157   return round(lightPercentage * 100.0) / 100.0; // Redondear a 2 decimales
158 }
```

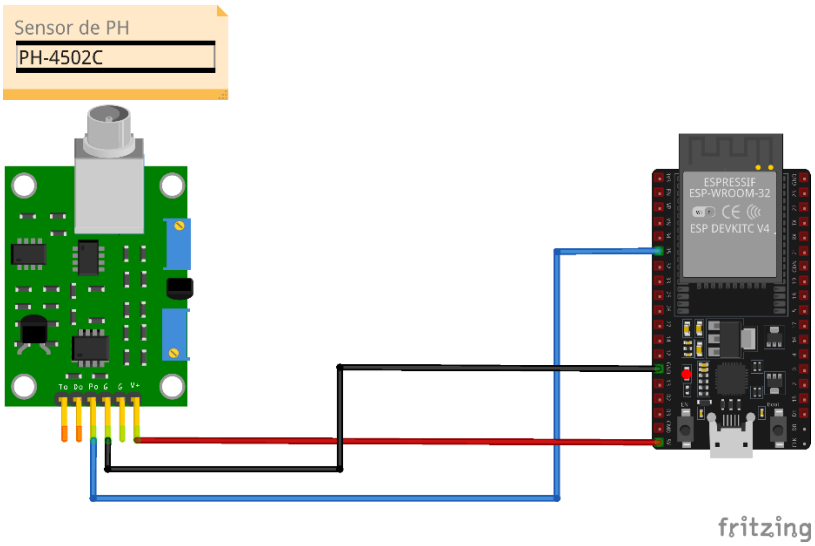
Nota: *Diseño de autoría propia.*

3.7.1.4 Configuración del sensor de pH

El sensor de pH ofrece distintas opciones de conexión; para este caso, se utilizó un pin GND conectado a la línea negativa del ESP32, mientras que el pin de señal se conectó al GPIO 35 del microcontrolador. Este mismo nodo se encuentra acoplado a una sonda que se sumerge en la solución nutritiva para la medición. En la **Figura 34** se presenta el esquema de conexión empleado para integrar el sensor con el sistema de adquisición de datos.

Figura 34

Conexiones del sensor de pH



Nota: *Diseño de autoría propia.*

El sensor entrega valores analógicos que son procesados mediante una ecuación de calibración basada en soluciones buffer de referencia con pH 4 y pH 7. Dicha calibración permite ajustar la respuesta del sensor para obtener mediciones precisas y confiables, empleando las siguientes relaciones:

$$m = \frac{7.0 - 4.0}{v_{cal4} + v_{cal7}} \quad \ldots (8)$$

$$b = 7.0 - (m * v_{cal7}) \quad \ldots (9)$$

$$PH = m * v_{esp} + b \quad \ldots (10)$$

Donde:

$m =$ Pendiente de la recta de calibración

$b =$ interseccion

$PH =$ Valor calculado de acidez del agua

v_{cal4} Y $v_{cal7=}$ Voltajes obtenidos durante la calibración con soluciones buffer pH 4 y pH 7

v_{esp}= Voltaje leído en el GPIO durante la medición

En el código presentado en la **Figura 35** se observa la implementación de este algoritmo, el cual permite el sensado en tiempo real para el posterior envío de los datos al Edge Gateway para su procesamiento y toma de decisiones.

Figura 35

Código implementado para sensor de pH

```
43  // Valores fijos de calibración (ajustar según tu sensor)
44  const float CAL_VOLTAGE_4 = 2.15;  // Voltaje para pH 4
45  const float CAL_VOLTAGE_7 = 1.80;  // Voltaje para pH 7
46
47  // Constantes para filtrado pH
48  #define PH_WINDOW_SIZE 11
49  #define PH_READING_INTERVAL 200
50  #define PH_STABILITY_SAMPLES 8
51  #define PH_MAX_VARIATION 0.15
152 // ========== MÉTODOS PARA PH ========== //
153
154 float voltageToPH(float voltage) {
155   if (voltage < 0 || voltage > 5.0) {
156     return lastStablePH;
157   }
158
159   float slope = (7.0 - 4.0) / (CAL_VOLTAGE_7 - CAL_VOLTAGE_4);
160   float intercept = 7.0 - slope * CAL_VOLTAGE_7;
161
162   float ph = slope * voltage + intercept;
163   return constrain(ph, 0.0, 14.0);
164 }
```

Nota: *Diseño de autoría propia.*

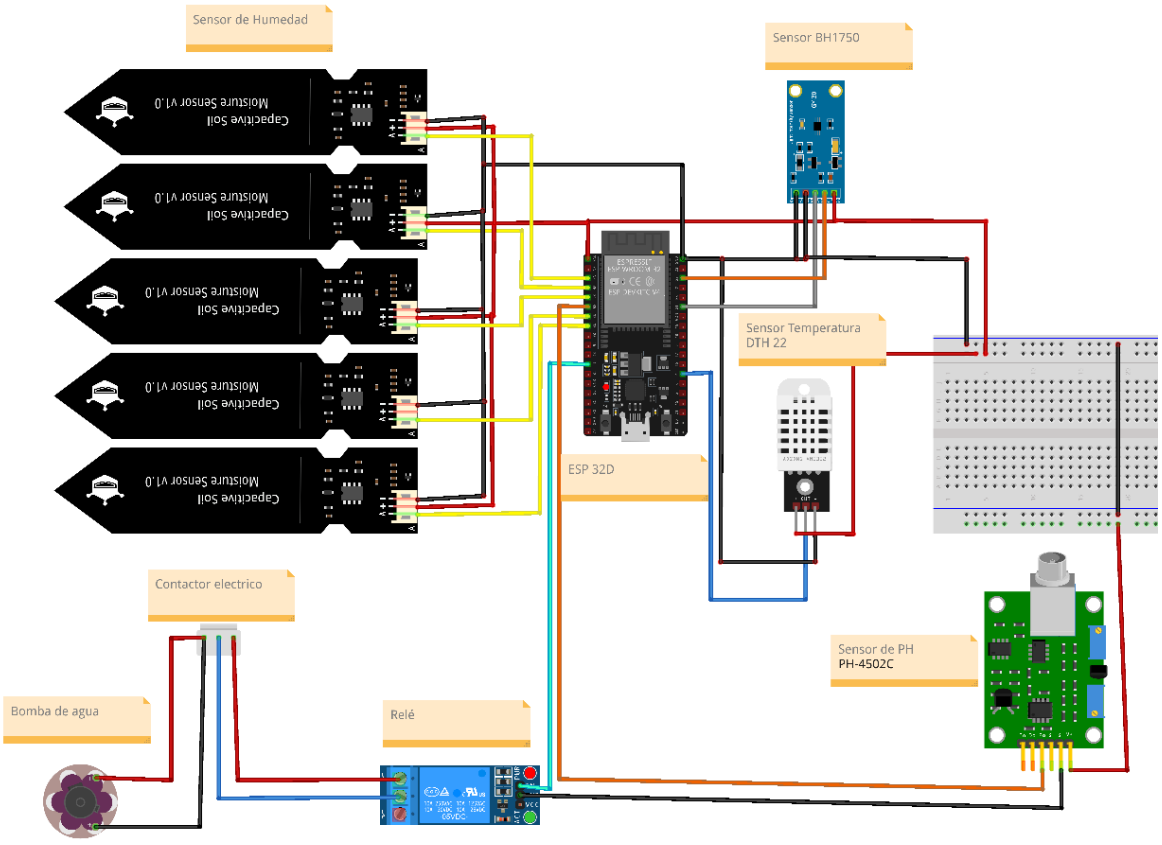
.

3.7.2 Configuración nodo sensor

Una vez establecidas las conexiones individuales y los códigos de cada sensor, se procedió a integrar todos los componentes en un diagrama unificado del nodo mostrado en la **Figura 36**. Como parte de esta integración, se asignó el pin GPIO 19 para el control de la bomba de riego, actuando como salida digital que responde a las decisiones del sistema. De manera paralela, los datos sensados (humedad, temperatura, pH y luminosidad) son procesados y enviados a la placa principal para su monitoreo y análisis, completando así el ciclo de automatización del sistema FVH.

Figura 36

Conexión final de los sensores a la placa ESP 32D



Nota: *Diseño de autoría propia.*

El código implementado en el nodo es la consolidación de todos los códigos y el mostrado en la **Figura 37**. corresponde empaquetamiento y envío de datos al servidor MQTT alojado en la Raspberry Pi 5, así como la activación de la bomba de riego mediante suscripción a los topics correspondientes. El sistema está configurado para ejecutarse en ciclos de 10 minutos, un intervalo equilibrado que permite un monitoreo constante sin saturar la red o los recursos del dispositivo, garantizando así un funcionamiento estable y eficiente de toda la infraestructura.

Figura 37

Código para el nodo sensor y actuador



```
// ===== CONFIGURACIÓN MQTT ===== //
const char* mqtt_topic_sensores = "sensor/data";
const char* mqtt_topic_control = "hidroponia/accion"; // Para recibir comandos
const char* mqtt_topic_estado = "hidroponia/estado";  // Para publicar estado

// ===== SETUP WIFI ===== //
void setup_wifi() {
  delay(100);
  Serial.println("Conectando a WiFi...");
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("Conectado a WiFi");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
}

// ===== CALLBACK MQTT ===== //
void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Mensaje recibido [");
  Serial.print(topic);
  Serial.print("]: ");

  String mensaje = "";
  for (int i = 0; i < length; i++) {
    mensaje += (char)payload[i];
  }
  Serial.println(mensaje);

  // Controlar electroválvula según el mensaje
  if (String(topic) == mqtt_topic_control) {
    if (mensaje == "ACTIVAR_RIEGO") {
      activarElectrovalvula();
    } else if (mensaje == "DETENER_RIEGO") {
      desactivarElectrovalvula();
    }
  }
}

  // Crear JSON
  String payload = "{";
  payload += "\"temperatura\":" + String(temperatura, 2) + ",";
  payload += "\"humedad_suelo\":" + String(humedadSuelo, 2) + ",";
  payload += "\"luz\":" + String(luz, 2);
  payload += "}";

  // Publicar
  if (client.publish(mqtt_topic_sensores, payload.c_str())) {
    Serial.println("✅ Datos publicados: " + payload);
  } else {
    Serial.println("❌ Error publicando datos");
  }
}
```

Nota: *Diseño de autoría propia.*

3.7.3 Configuración de Edge Gateway

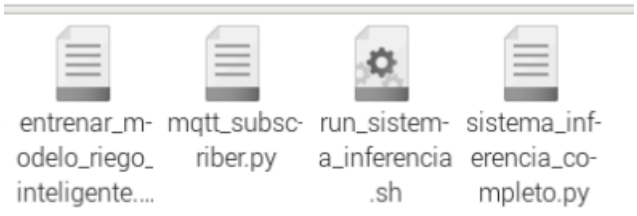
Esta sección detalla la configuración implementada en la placa principal del sistema (Raspberry Pi), la cual funciona como servidor de servicios y puerta de enlace (gateway) para la gestión y distribución de datos. Su propósito fundamental es centralizar la información y permitir su visualización en las distintas plataformas de monitoreo diseñadas para el usuario final. La configuración se estructura en las siguientes etapas: preparación del entorno virtual, entrenamiento del árbol de decisión, configuración del servidor MQTT, ejecución del sistema de inferencia e implementación del sistema de almacenamiento de información.

3.7.3.1 Programación

En la placa principal se implementó inicialmente un entorno virtual de Python para gestionar las dependencias específicas requeridas por los archivos del sistema las mismas que se observan en la **Figura 38**. Esta configuración asegura un ecosistema aislado y consistente para la ejecución del software. Posteriormente, se desplegaron los archivos de configuración necesarios para habilitar en la placa las funcionalidades de comunicación, control y toma de decisiones que constituyen el núcleo operativo del sistema.

Figura 38

Archivos de configuración de la placa central



Nota: *Diseño de autoría propia.*

3.7.3.1.1 Configuración MQTT

Primeramente, se establece el archivo de configuración para las conexiones del servidor MQTT, permitiendo que el nodo publique datos hacia la Raspberry Pi. Como se evidencia en la **Figura 39**, este archivo importa las librerías necesarias, entre las que se incluyen paho.mqtt.client y sqlite3 - previamente instalada para la gestión de la base de datos. El script crea la base de datos datos_sensores.db, verifica o genera la tabla sensores destinada al almacenamiento de las lecturas de los sensores, y configura los mecanismos de envío y recepción de mensajes para el control e intercambio de información según los tópicos definidos en el sistema.

Figura 39

Archivo de subscriptor de servicio MQTT

mqtt_subscriber.py | entrenar_model..._inteligente.py | sistema_inferencia_completo.py

```
import paho.mqtt.client as mqtt
import sqlite3
import json
from datetime import datetime

DB_NAME = "/media/erik/CUASQUER_E/base de datos/datos_sensores.db"
MQTT_TOPIC = "sensor/data"

def crear_tablas():
    """Crea las tablas necesarias - SIN columnas extra"""
    try:
        conn = sqlite3.connect(DB_NAME)
        cursor = conn.cursor()

        cursor.execute("SELECT name FROM sqlite_master WHERE type='table' AND name='sensores'")
        if not cursor.fetchone():
            cursor.execute('''CREATE TABLE sensores (
                              id INTEGER PRIMARY KEY AUTOINCREMENT,
                              timestamp DATETIME DEFAULT (datetime('now', 'localtime')),
                              temperatura REAL,
                              humedad_suelo REAL,
                              luz REAL,
                              ph REAL,
                              decision_riego TEXT,
                              minutos_riego INTEGER,
                              accion_ejecutada TEXT)''')
            print("✅ Tabla 'sensores' creada")

        cursor.execute('''CREATE TABLE IF NOT EXISTS eventos_riego (
                          id INTEGER PRIMARY KEY AUTOINCREMENT,
                          timestamp DATETIME DEFAULT (datetime('now', 'localtime')),
                          evento TEXT,
                          descripcion TEXT)''')

        conn.commit()
        conn.close()
        print("✅ Tablas verificadas")
    except Exception as e:
        print(f"⚠ Error: {e}")

def guardar_datos_sensores(temperatura, humedad_suelo, luz, ph):
    """Guarda SOLO los datos disponibles"""
    try:
        conn = sqlite3.connect(DB_NAME)
        cursor = conn.cursor()

        timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")

        cursor.execute("""INSERT INTO sensores
```

Nota: *Diseño de autoría propia.*

3.7.3.1.2 Entrenamiento del Modelo de Árbol de Decisión

El proceso de entrenamiento del primer modelo predictivo se implementó mediante un archivo de configuración en Python que ejecuta las siguientes etapas:

A. Importación y Configuración Inicial

El script inicia con la importación de librerías esenciales para el procesamiento de datos (pandas), machine learning (scklearn) y serialización del modelo (joblib). Se configura el entorno suprimiendo advertencias específicas de sklearn para mantener la claridad en los logs, y se definen rutas absolutas para garantizar el acceso correcto a los archivos de datos y el modelo entrenado, asegurando portabilidad entre diferentes sistemas. Esta etapa sienta las bases técnicas para la reproducibilidad y escalabilidad del proceso.

Figura 40

Librerías necesarias para el árbol de decisión e Inicialización



Nota: *Diseño de autoría propia.*

B. Carga y Procesamiento de Datos

En esta sección se presenta la extracción, transformación y limpieza de los registros vectoriales obtenidos de los sensores IoT registrados y estructurados en formato CSV. La función cargar_datos_csv() opera como un mecanismo de respaldo algorítmico (fallback) que garantiza la continuidad operativa del sistema en caso de desconexión con la base de datos principal o el bróker de mensajería. Metodológicamente, el script implementa un bloque de control de excepciones robusto que valida la integridad estructural de la matriz de datos, verificando la presencia obligatoria de las variables explicativas clave: temperatura, humedad del suelo, luminosidad y pH. Asimismo, se efectúa un proceso de normalización temporal al parsear cadenas de caracteres a objetos indexables de tipo datetime, aislando vectorialmente la variable horaria (dt.hour) o asignando un valor por defecto determinista en ausencia de esta, sentando así las bases analíticas requeridas por los modelos predictivos subsiguientes. Adicionalmente, todo esto mostrándose en la **Figura 41**.

Figura 41

Carga datos para entrenamiento

```
166
167 def cargar_datos_csv():
168     """
169     Carga los datos desde el archivo CSV (fallback)
170     """
171     try:
172         if not os.path.exists(ruta_csv):
173             print(f"⚠ Archivo CSV no encontrado: {ruta_csv}")
174             return None
175
176         df = pd.read_csv(ruta_csv)
177         print(f"✅ Cargados {len(df)} registros desde CSV")
178
179         # Verificar columnas necesarias
180         columnas_necesarias = ['temperatura', 'humedad_suelo', 'luz', 'ph']
181         columnas_faltantes = [col for col in columnas_necesarias if col not in df.columns]
182
183         if columnas_faltantes:
184             print(f"❌ Columnas faltantes en CSV: {columnas_faltantes}")
185             return None
186
187         # Convertir timestamp si existe
188         if 'timestamp' in df.columns:
189             df['timestamp'] = pd.to_datetime(df['timestamp'])
190             df['hora'] = df['timestamp'].dt.hour
191         else:
192             df['hora'] = 12  # valor por defecto
193
194         return df
195
196     except Exception as e:
197         print(f"❌ Error cargando CSV: {e}")
198         return None
```

Nota: *Diseño de autoría propia*

C. Definición de Reglas de Negocio Inteligentes

Se implementa una función definir_decision_inteligente mostrada en la **Figura 42**, Esta ilustración detalla la estructura formal del motor de inferencia heurístico configurado para el control de actuadores en el sistema de Forraje Verde Hidropónico (FVH). El código implementa una arquitectura jerárquica de toma de decisiones segmentada en tres niveles operacionales críticos: Protección (Nivel 1), Control de Humedad (Nivel 2) y Momentos Óptimos (Nivel 3). Mediante un enfoque determinista basado en condicionales anidados, el algoritmo evalúa en tiempo real variables micrometeorológicas y edafoclimáticas esenciales como la temperatura superficial, el porcentaje de humedad relativa o del sustrato, y el

potencial de hidrógeno (pH). Cabe destacar que el Nivel 1 prioriza la resiliencia biológica del cultivo ante escenarios de estrés abiótico extremo (v.g., sequía crítica, heladas severas, olas de calor y saturación hídrica), actuando como un sistema de interrupción segura (fail-safe) que inhibe o activa el riego de manera mandatoria para resguardar la integridad fitosanitaria antes de ceder el control a los módulos de optimización agronómica convencionales.

Figura 42

Reglas para el entrenamiento del árbol.

```
        # ===== NIVEL 1 - PROTECCIÓN ===== #

        # Regla 1 – Sequía crítica
        if humedad < 15:
            if cls.PH_OPTIMO_MIN <= ph <= cls.PH_OPTIMO_MAX:
                return 3  # REGAR 5 minutos
            else:
                return 0  # NO REGAR (pH malo)

        # Regla 2 – Helada severa
        if temp < 5:
            return 0  # NO REGAR

        # Regla 3 – Frío con humedad alta
        if temp < 12 and humedad > 60:
            return 0  # NO REGAR

        # Regla 4 – Ola de calor extrema
        if temp > 35:
            return 0  # NO REGAR

        # Regla 5 – Saturación
        if humedad > 70:
            return 0  # NO REGAR

        # ===== NIVEL 2 - CONTROL DE HUMEDAD ===== #

        # Regla 6 – Zona óptima
        if 55 <= humedad <= 65 and cls.PH_OPTIMO_MIN <= ph <= cls.PH_OPTIMO_MAX:
            return 0  # NO REGAR

        # ===== NIVEL 3 - MOMENTOS ÓPTIMOS ===== #

        # Regla 7 – Déficit moderado
        if humedad < 45:
            if cls.PH_OPTIMO_MIN <= ph <= cls.PH_OPTIMO_MAX:
                return 2 if humedad < 30 else 1  # 3min si <30%, 1min si 30-45%

        # Regla 8 – Riego nocturno
        if 0 <= hora <= 5 and humedad < 50:
            if cls.PH_OPTIMO_MIN <= ph <= cls.PH_OPTIMO_MAX:
                return 2  # 3 minutos

        # Regla 9 – Mañana fresca
        if 5 < hora <= 9 and humedad < 55:
            if cls.PH_OPTIMO_MIN <= ph <= cls.PH_OPTIMO_MAX:
```

Nota: *Diseño de autoría propia.*

D. Preparación para Machine Learning

Los datos procesados se dividen en conjuntos de entrenamiento (80%) y prueba (20%) mediante train_test_split, utilizando una semilla aleatoria (random_state=42) para garantizar reproducibilidad. Las características seleccionadas (humedad_suelo, temperatura, luz) y la variable objetivo (regar) se estructuran para el aprendizaje supervisado, previo análisis de distribución de clases para verificar balanceo de decisiones.

E. Entrenamiento y Evaluación del Modelo

Se entrena un clasificador de Árbol de Decisión con hiperparámetros optimizados (max_depth=6, min_samples_split=15, min_samples_leaf=8) para equilibrar complejidad y generalización como se muestra en la **Figura 43**. La evaluación incluye cálculo de precisión en el conjunto de prueba y exportación de reglas aprendidas mediante export_text, proporcionando transparencia al proceso de decisión. El modelo se serializa con joblib para su implementación, y se valida con casos de ejemplo que simulan escenarios típicos, demostrando su capacidad predictiva en condiciones variables.

Figura 43

Entrenamiento del árbol de decisión.

```
354
355     # Preparar datos con etiquetas
356     df = preparar_datos_entrenamiento(df)
357
358     # Seleccionar características y etiquetas
359     features = ['temperatura', 'humedad_suelo', 'luz', 'ph']
360     X = df[features]
361     y = df['accion_numerica']
362
363     # Dividir en entrenamiento y prueba
364     X_train, X_test, y_train, y_test = train_test_split(
365         X, y, test_size=0.2, random_state=42, stratify=y
366     )
367
368     # Crear modelo con parámetros optimizados
369     modelo = DecisionTreeClassifier(
370         max_depth=6,
371         min_samples_split=15,
372         min_samples_leaf=8,
373         random_state=42,
374         class_weight='balanced'
375     )
376
377     # Entrenar
378     print("\n⚙  Entrenando modelo...")
379     modelo.fit(X_train, y_train)
380
381     # Evaluar
382     y_pred = modelo.predict(X_test)
383     accuracy = accuracy_score(y_test, y_pred)
384
385     # Validación cruzada
386     cv_scores = cross_val_score(modelo, X, y, cv=5)
387
```

Nota: *Diseño de autoría propia.*

F. Validación y Ejemplos de Predicción

Finalmente, la **Figura 44** ilustra el entorno analítico diseñado para la validación funcional, el control de calidad empírico y el testeo unitario del modelo de aprendizaje supervisado previamente entrenado. Mediante la función de diagnóstico probar_modelo_con_ejemplos(), se somete al clasificador a un banco de pruebas multivariable compuesto por vectores de datos sintéticos y reales que simulan las contingencias agronómicas más representativas del ecosistema FVH (tales como "Sequía crítica", "Helada", "Saturación" y "Zona óptima"). El bucle de prueba transforma cada diccionario de variables en un DataFrame compatible con

el modelo, calculando no solo la clase discreta predicha de acuerdo con la escala operativa estandarizada (v.g., NO_REGAR, REGAR_1min, etc.), sino también el vector de probabilidad posterior asociado mediante predict_proba(). Este procedimiento permite cuantificar el nivel de certeza epistémica del árbol de decisión frente a transiciones de estado ambientales complejas, validando la precisión adaptativa del sistema IoT antes de su despliegue físico en el invernadero automatizado.

Figura 44

Ejemplos de predicción

```python
def probar_modelo_con_ejemplos(modelo):
    """
    Prueba el modelo con ejemplos concretos
    """
    print("\n" + "="*60)
    print("🔬 PROBANDO MODELO CON EJEMPLOS")
    print("="*60)

    ejemplos = [
        {"desc": "Sequia critica", "temp": 25, "hum": 10, "luz": 50, "ph": 6.0},
        {"desc": "Sequia con pH malo", "temp": 25, "hum": 10, "luz": 50, "ph": 8.5},
        {"desc": "Helada", "temp": 3, "hum": 40, "luz": 20, "ph": 6.2},
        {"desc": "Déficit moderado", "temp": 22, "hum": 35, "luz": 60, "ph": 6.0},
        {"desc": "Noche", "temp": 18, "hum": 45, "luz": 5, "ph": 6.1},
        {"desc": "Mañana fresca", "temp": 20, "hum": 50, "luz": 40, "ph": 6.0},
        {"desc": "Preparación calor", "temp": 30, "hum": 45, "luz": 85, "ph": 6.2},
        {"desc": "Zona óptima", "temp": 24, "hum": 60, "luz": 70, "ph": 6.0},
        {"desc": "Saturación", "temp": 22, "hum": 80, "luz": 50, "ph": 6.3},
        {"desc": "pH ácido", "temp": 25, "hum": 40, "luz": 60, "ph": 4.5},
        {"desc": "pH alcalino", "temp": 25, "hum": 40, "luz": 60, "ph": 7.8},
        {"desc": "Calor extremo", "temp": 38, "hum": 30, "luz": 90, "ph": 6.0},
    ]

    clases = ['NO_REGAR', 'REGAR_1min', 'REGAR_3min', 'REGAR_5min']

    for ej in ejemplos:
        X_test = pd.DataFrame([[ej['temp'], ej['hum'], ej['luz'], ej['ph']]],
                              columns=['temperatura', 'humedad_suelo', 'luz', 'ph'])
        pred = modelo.predict(X_test)[0]
        proba = modelo.predict_proba(X_test)[0]

        print(f"\n🔹 {ej['desc']}:")
        print(f"   T={ej['temp']}°C, H={ej['hum']}%, L={ej['luz']}%, pH={ej['ph']}")
        print(f"   → {clases[pred]} (confianza: {proba[pred]:.2f})")
```

Nota: *Diseño de autoría propia.*

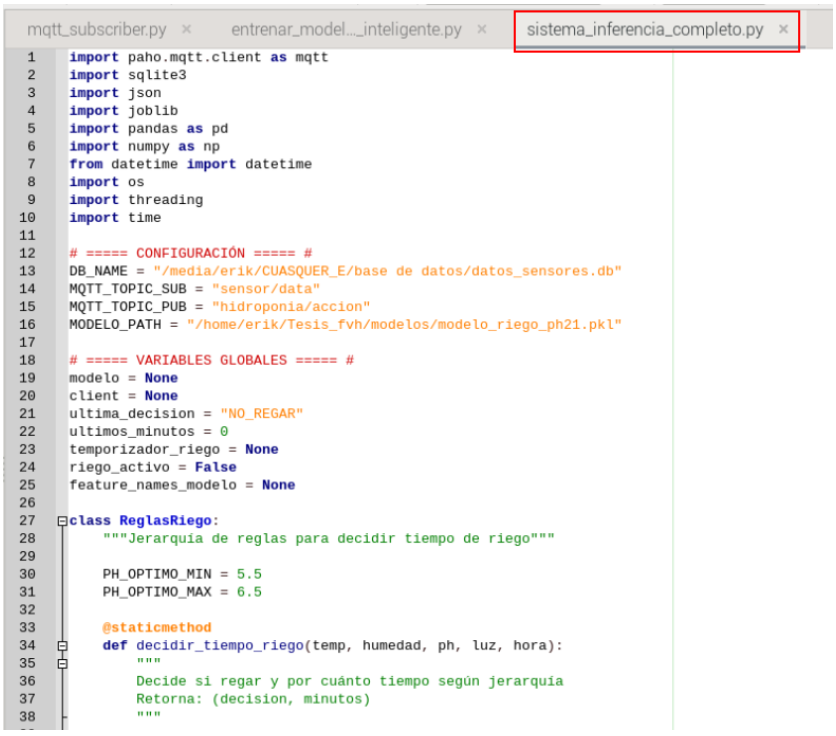
.

3.7.3.1.3 Ejecución del sistema de inferencia

Una vez entrenado el árbol de decisión, se despliega el sistema de inferencia mediante un archivo de configuración ejecutable mismo que se muestra en la **Figura 45**. Este sistema inicializa los parámetros operativos fundamentales: la base de datos SQLite para el almacenamiento local, los tópicos MQTT para la comunicación bidireccional (sensor/data para recepción e hidroponia/accion para el envío de comandos), y la carga del modelo de machine learning. En su núcleo, el módulo de base de datos garantiza la trazabilidad completa del sistema, registrando en una estructura normalizada cada timestamp, las lecturas de los sensores, la decisión de riego generada por el modelo y la acción final ejecutada, lo que facilita la auditoría y el análisis posterior del rendimiento.

Figura 45

Archivo del sistema de inferencia



Nota: *Diseño de autoría propia.*

3.7.3.1.4 Control de Actuadores y Publicación MQTT

El archivo de inferencia también ejecuta el control del sistema una vez traduce las decisiones en comandos operativos ("ACTIVAR_RIEGO"/"DETENER_RIEGO") como se muestra en la **Figura 46**, actualizando el estado global y publicando indefectiblemente mediante publicar_mqtt_accion. Este diseño asegura que cada decisión se refleje en un comando físico, eliminando estados intermedios o ambiguos y manteniendo sincronización entre la lógica de control y el actuador real.

Figura 46

Código para el control mediante mensajes en MQTT

```
def gestionar_tiempo_riego(minutos):
    """Gestiona el temporizador de riego"""
    global riego_activo, temporizador_riego

    if riego_activo and temporizador_riego:
        print("[] Deteniendo riego anterior para iniciar nuevo ciclo")
        detener_riego_actual()

    print(f"[] INICIANDO RIEGO por {minutos} minuto(s)")
    client.publish(MQTT_TOPIC_PUB, "ACTIVAR_RIEGO")
    riego_activo = True

    guardar_evento_riego("INICIO_RIEGO", f"Duración: {minutos} minutos")

    segundos = minutos * 60
    temporizador_riego = threading.Timer(segundos, detener_riego_actual)
    temporizador_riego.daemon = True
    temporizador_riego.start()

    print(f"[] Riego programado por {minutos} minuto(s) ({segundos} segundos)")

def detener_riego_actual():
    """Detiene el riego actual y publica comando"""
    global riego_activo, temporizador_riego

    if riego_activo:
        print("\n[] TIEMPO COMPLETADO - Deteniendo riego")
        client.publish(MQTT_TOPIC_PUB, "DETENER_RIEGO")
        riego_activo = False
        temporizador_riego = None
        guardar_evento_riego("FINALIZADO_POR_TIEMPO", "Riego completado")
    else:
        print("⚠ No hay riego activo para detener")
```

Nota: *Diseño de autoría propia.*

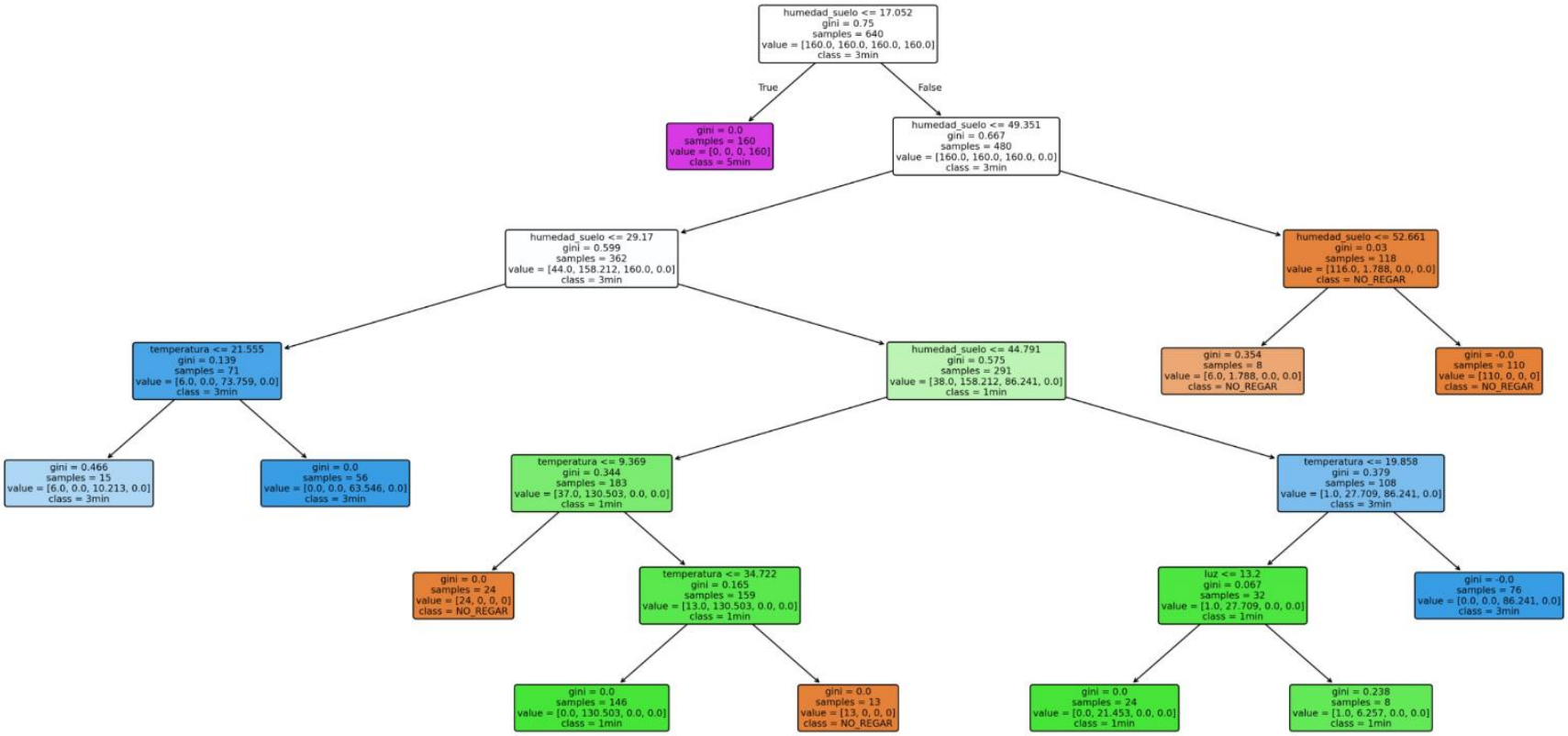
3.7.4 Árbol de decisión

Una vez expuestos los archivos de configuración, se muestra el estado del árbol de decisión entrenado. Cada nodo interno del árbol contiene una condición basada en una característica del conjunto de datos (como humedad del suelo, luz o temperatura), mientras que el valor de Gini indica el nivel de pureza o mezcla de las clases en ese punto. Los nodos terminales (hojas) muestran la clase final predicha, en este caso REGAR o NO REGAR, junto con el número de muestras que llegaron a dicha hoja. Este árbol evidencia que la variable con mayor poder discriminante es la humedad del suelo, seguida por la intensidad de luz, lo que coincide con la lógica agronómica del proceso de riego. En conjunto, la estructura visual del árbol permite interpretar las reglas que el modelo aprendió a partir de los datos históricos generados por los sensores del sistema IoT.

La parte visual del árbol es expresada en la **Figura 47**.

Figura 47

Visualización del Árbol de decisión entrenado

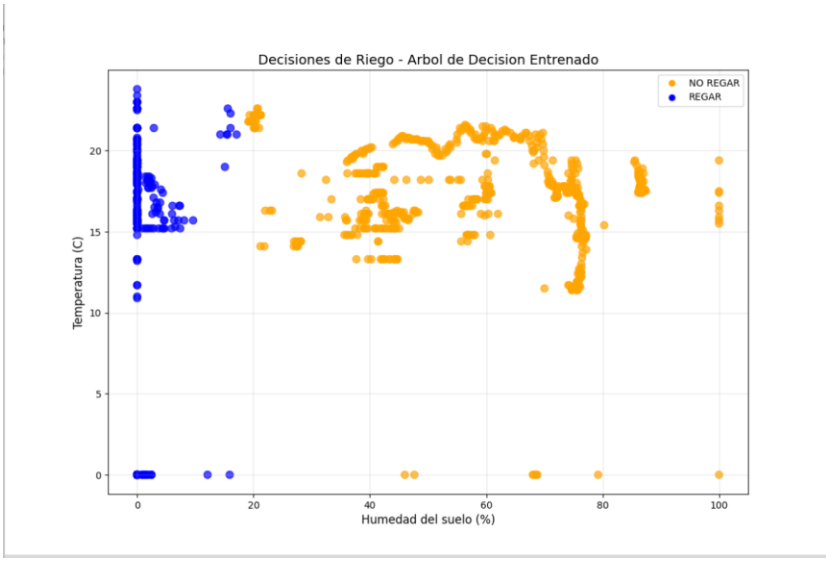


Nota: *Diseño de autoría propia.*

El gráfico de dispersión mostrado en la **Figura 48** ilustra la distribución de los datos empleados para entrenar el modelo de decisión y muestra visualmente las predicciones realizadas por el árbol. En el eje X se encuentra la humedad del suelo (%) y en el eje Y la temperatura ambiental (°C), mientras que los colores diferencian las dos clases output: REGAR (puntos naranjas) y NO REGAR (puntos azules). Esta representación permite identificar de manera clara las zonas de decisión del modelo, observando que los valores de humedad baja tienden a resultar en acciones de riego, mientras que valores altos, especialmente combinados con temperaturas elevadas o altos niveles de luz, corresponden a decisiones de no riego. El gráfico complementa la interpretabilidad del modelo al mostrar cómo se distribuyen los escenarios reales en el espacio de características.

Figura 48

Gráfico de dispersión respecto a decisiones.



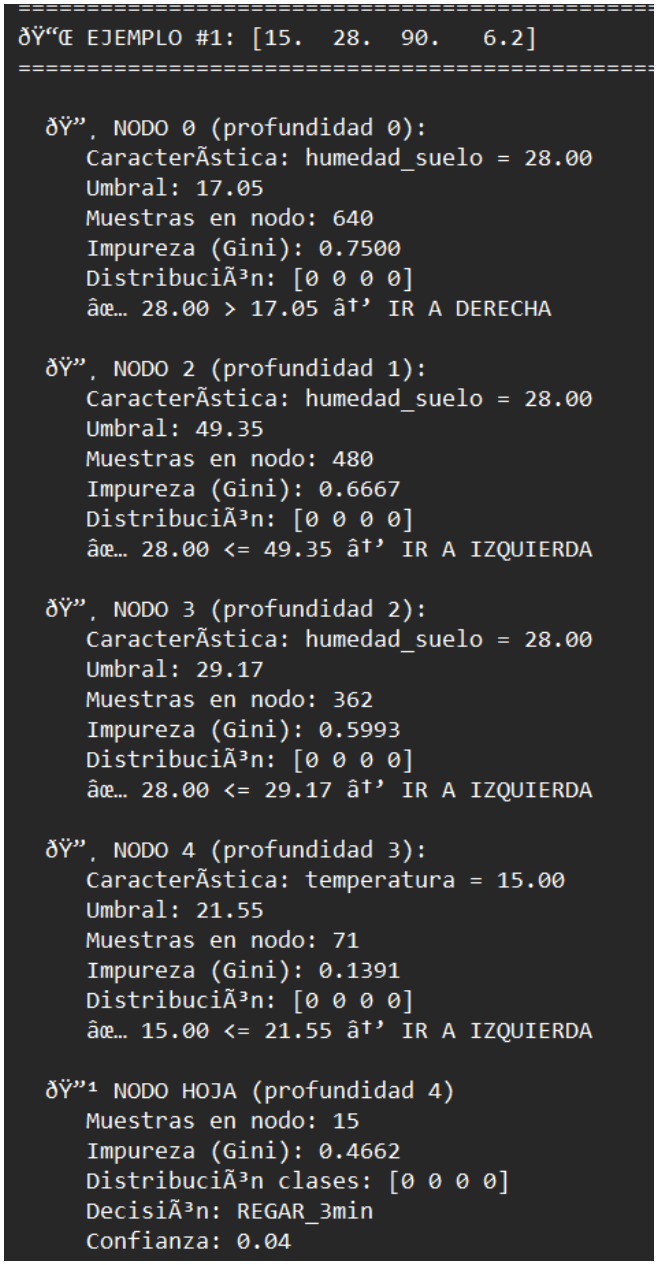
Nota: *Diseño de autoría propia.*

Los resultados en consola como se observa en la **Figura 46**, muestran paso a paso la ruta que cada ejemplo de entrada sigue dentro del árbol de decisión hasta llegar a la hoja final. Para

cada caso de prueba se listan las condiciones evaluadas, la dirección tomada en cada nodo (izquierda o derecha), y finalmente la clase asignada por el modelo.

Figura 49

Rutas de decisión



Nota: *Diseño de autoría propia.*

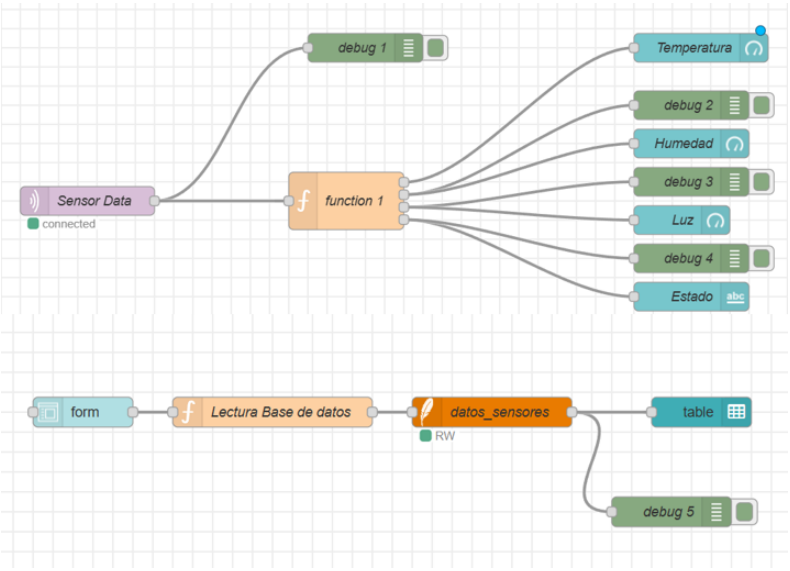
Este análisis es valioso porque proporciona transparencia al sistema de riego inteligente, evidenciando cómo una combinación específica de humedad, temperatura y luz activa un conjunto particular de reglas aprendidas. La trazabilidad de la decisión permite validar que las predicciones del modelo sean coherentes con la lógica del dominio agrícola y con las reglas inteligentes iniciales empleadas para generar las etiquetas del dataset.

3.7.5 Plataforma monitoreo

En cuanto a la plataforma de monitoreo, se implementó Node-RED para visualizar los datos publicados por el nodo en un dashboard especializado. Los datos recibidos pasan through una función de procesamiento que los adapta para su correcta visualización en los distintos componentes de la interfaz. Paralelamente, se integra la base de datos SQLite para recuperar y mostrar el historial de registros en el dashboard, permitiendo el análisis temporal del comportamiento del sistema.

Figura 50

Nodos de configuración de la plataforma de monitoreo

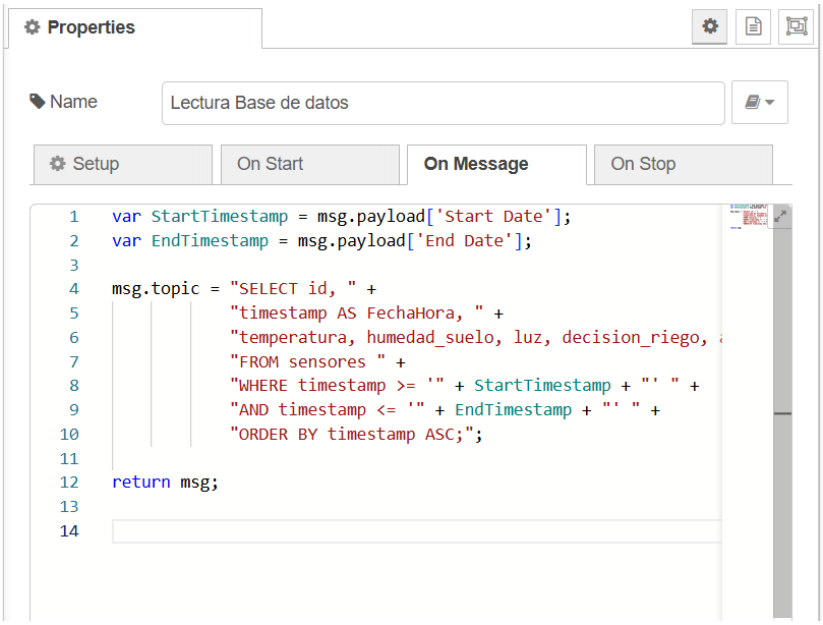


Nota: *Diseño de autoría propia.*

De este modo, se establece la conexión con la base de datos para recuperar y visualizar los valores almacenados en la tabla sensores mediante el código en el nodo de la **Figura 51**, la cual centraliza toda la información registrada por el sistema. Esta consulta permite la exposición integral de los datos para su posterior análisis y monitoreo.

Figura 51

Extracción de datos de la base de datos

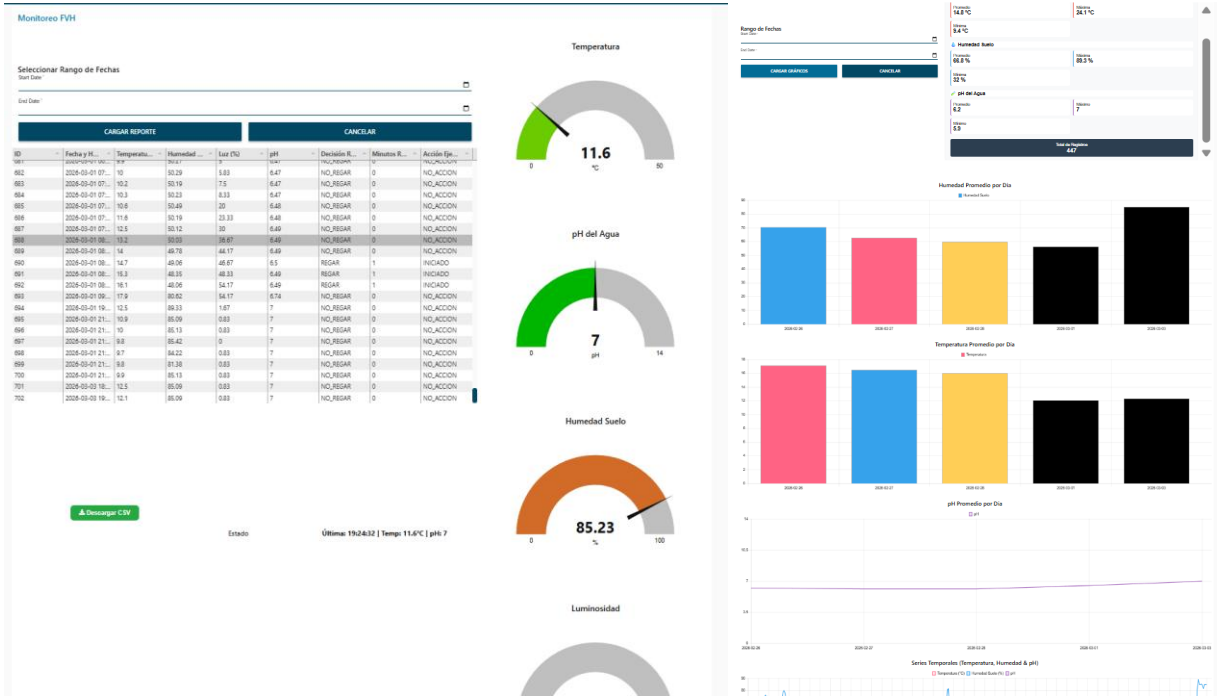


Nota: *Diseño de autoría propia.*

Finalmente, se presenta la interfaz de visualización destinada al usuario final en la **Figura 52**, donde podrá observar en tiempo real tanto el almacenamiento como la captura de datos realizada por los sensores del sistema.

Figura 52

Dashboard final en la plataforma de node-red



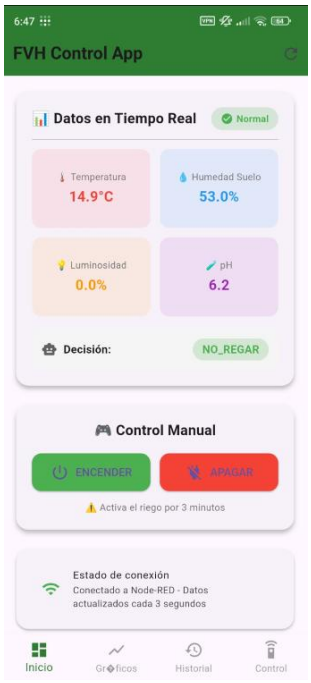
Nota: *Diseño de autoría propia.*

3.8 Aplicación móvil (FVH control app)

Con el propósito de brindar al usuario final (la criadora de cuyes) una herramienta accesible para monitorear y, en caso necesario, activar el sistema de riego de forma remota, se desarrolló una aplicación móvil denominada FVH Control App. Esta aplicación se enlaza con el dashboard de Node-RED mediante el protocolo HTTP/REST como mecanismo principal de comunicación, el cual permite realizar solicitudes para obtener el objeto JSON generado por el nodo sensor y transmitido al Edge Gateway, logrando que los datos lleguen de manera actualizada al dispositivo móvil de la usuaria. La información presentada corresponde a los registros almacenados en la base de datos del sistema. La interfaz general de la aplicación se muestra en la **Figura 53**.

Figura 53

Interfaz App móvil



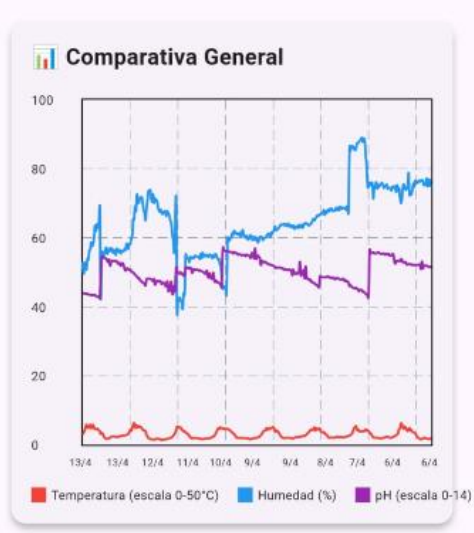
Nota: *Diseño de autoría propia.*

Adicionalmente, la aplicación realiza peticiones para acceder a los datos requeridos por el

ingeniero agrónomo, presentando diagramas de variación de las variables monitoreadas mediante series temporales generadas a partir de los registros del sistema, tal como se observa en la **Figura 54**.

Figura 54

Series temporales de los datos de los sensores



Nota: *Diseño de autoría propia.*

En la sección de historial, el usuario puede interactuar con la base de datos SQLite a través de consultas realizadas mediante comunicación HTTP/REST. Esta funcionalidad permite buscar registros según diferentes criterios, diferenciando entre las activaciones del riego decididas automáticamente por el árbol de decisión, las activaciones manuales y los momentos en que el sistema determinó no activar el riego, evidenciándose esto en la **Figura 55**.

Figura 55

Historial de registros en app móvil



Nota: *Diseño de autoría propia.*

3.9 VPN

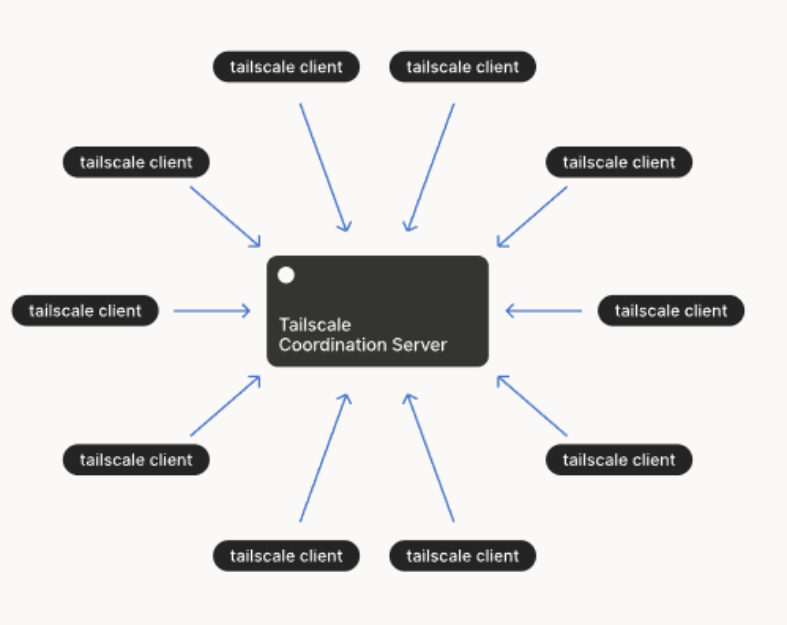
Para posibilitar el monitoreo remoto desde la aplicación móvil, se implementó Tailscale como solución de red privada virtual basada en WireGuard y en el modelo de confianza cero. Esta herramienta establece un canal cifrado y directo entre el dispositivo móvil y el servidor Node-RED, asignando una dirección fija virtual como en nuestro caso (100.108.50.80), sin requerir una IP pública estática, lo que resuelve las limitaciones de conectividad en entornos domésticos o institucionales.

Desde el punto de vista técnico, Tailscale genera una red superpuesta con direcciones CGNAT estables, emplea perforación NAT mediante servidores STUN y recurre a retransmisión DERP en caso de bloqueo. Su cifrado simétrico ChaCha20 y autenticación

Poly1305, junto con el intercambio de claves Curve25519, aportan baja latencia y escaso consumo de recursos. La VPN se instaló tanto en los dispositivos estableciéndose como clientes tailscale como se muestra en la **Figura 56**, con la aplicación móvil como en aquellos que acceden vía web al dashboard de Node-RED, asegurando la confidencialidad e integridad de los datos transmitidos.

Figura 56

Tailscale estructura



Nota: *Diseño de autoría propia.*

CAPÍTULO IV

IMPLEMENTACIÓN DEL SISTEMA

Este capítulo describe las fases para la transición del prototipo funcional a un sistema a escala real destinado a la producción continua de Forraje Verde Hidropónico (FVH). Se inicia con la construcción y adecuación del espacio físico, incluyendo la instalación de estructuras de soporte, el sistema de riego por aspersión y los canales de drenaje, seguido de la implementación física de sensores, actuadores y cableado eléctrico que permitirán la automatización integral del proceso de cultivo.

Paralelamente, se establece un plan de pruebas y siembra periódica, diseñado para validar el funcionamiento del sistema, verificar el cumplimiento de los objetivos iniciales y alcanzar progresivamente el 100% de productividad. Este plan incluye protocolos de monitoreo, evaluación de rendimiento y ajustes iterativos basados en los datos obtenidos.

Finalmente, se define un cronograma de mantenimiento preventivo y se configuran alertas automatizadas que notificarán al usuario sobre tareas de mantenimiento necesarias, garantizando la sostenibilidad y operatividad del sistema a largo plazo, aumentando su vida útil.

4.1 Implementación

En la fase de implementación, se construyó y desplegó el sistema IoT basado en los diseños previamente establecidos, utilizando componentes como la placa ESP32 (como nodo principal), la Raspberry Pi 5 (como gateway) y sensores especializados (DS18B20 para temperatura, FC-28 para humedad). Se integró el sistema de riego automatizado y se configuró el software de control, asegurando que toda la infraestructura funcionara de manera coordinada dentro del invernadero destinado al cultivo de FVH.

4.1.1 Construcción del invernadero para FVH.

La implementación del sistema requirió la adecuación de un espacio específicamente diseñado para el desarrollo del forraje hidropónico, garantizando condiciones climáticas controladas y protección frente a agentes externos y animales que pudieran afectar las estructuras o el cultivo. Este proceso se llevó a cabo con el asesoramiento de un profesional del área de construcción de invernaderos y el apoyo de dos asistentes, entre ellos el autor de este proyecto.

Los trabajos de construcción se extendieron durante dos días, utilizando materiales especializados para invernaderos mismos que se observan en la **Figura 57**. Las dimensiones finales del espacio son de 5 m × 5 m (ancho × largo), con una altura de 2,4 m en los laterales y 2,8 m en la cumbrera, lo que permite una adecuada evacuación del agua en caso de lluvias y facilita la circulación del aire en el interior del ambiente controlado.

Figura 57

Construcción del Invernadero para desarrollo de FVH.



Nota: *Imagen de autoría propia.*

4.1.2 Adquisición de la bomba de agua para el riego

La bomba de agua fue seleccionada en función de los requerimientos específicos del sistema de riego por aspersión, optándose por un modelo centrífugo que garantiza una presión constante y uniforme en toda la red de distribución. Esta bomba posee una potencia de 1 HP (caballo de fuerza) y opera a un voltaje de alimentación de 110 V misma que podemos observar en la **Figura 58**, proporcionando el caudal y la presión necesarios para asegurar una cobertura homogénea del agua en todas las bandejas de cultivo del sistema

FVH.

Figura 58

Bomba de riego Paolo



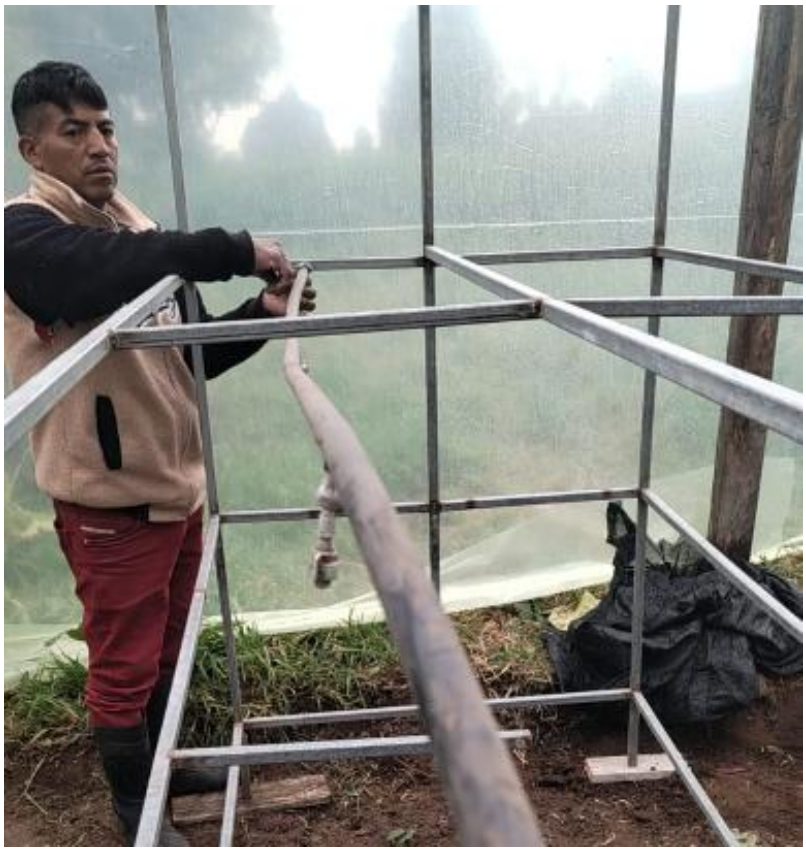
Nota: *Imagen de autoría propia.*

4.1.3 Implementación de los aspersores en las estructuras.

Una vez adquiridos todos los componentes, se procedió a la instalación de los aspersores en los estantes diseñados para este fin como podemos notar en la **Figura 59**. Para garantizar un riego uniforme y eficiente, cada aspersor fue nivelado cuidadosamente en todos los ejes, asegurando una cobertura óptima del agua en cada bandeja de cultivo. Dada la subdivisión de los estantes en niveles y bloques independientes, los aspersores fueron distribuidos estratégicamente, asignando una unidad por cada sub-bloque. Esta configuración permite que el sistema de riego cubra de manera completa y homogénea toda el área prevista para la producción de forraje.

Figura 59

Implementación de microaspersores



Nota: *Imagen de autoría propia.*

Para esta etapa, se contó con la asistencia de un especialista en sistemas de riego, a quien se le presentó el diseño previamente planificado. Bajo su supervisión, se realizó la instalación y conexión de las mangueras de distribución, asegurando que la presión del agua se mantuviera uniforme en todos los tramos del sistema. Esta configuración garantiza que cada microaspersor reciba el caudal adecuado para operar de manera óptima, proporcionando así un riego homogéneo en todos los niveles de los estantes del cultivo, tal como se aprecia en la **Figura 60.**

Figura 60

Distribución de mangueras para microaspersión



Nota: *Imagen de autoría propia.*

Las mangueras de distribución se conectan directamente a la bomba adquirida (Paolo PC-122), la cual realiza la succión e impulsión del agua a la presión necesaria para el correcto funcionamiento del sistema de microaspersión. Este componente es el encargado de garantizar el caudal y la presión requeridos en toda la red de distribución, asegurando un riego uniforme en cada uno de los niveles de los estantes de cultivo, tal como se observa en la **Figura 61**.

Figura 61

Conexión de mangueras a la bomba



Nota: *Imagen de autoría propia.*

El tanque de almacenamiento está conectado a la red hídrica principal de la instalación y cuenta con un método de llenado automático mediante una válvula de flotador (boya). Este mecanismo se activa cuando el nivel del agua desciende, permitiendo el reabastecimiento automático del tanque. De esta manera, se garantiza que la bomba centrífuga siempre disponga del volumen de agua suficiente para su operación continua, previniendo daños por cavitación o funcionamiento en seco.

En la **Figura 62** podemos ver como esta visualizado el tanque ya instalado dentro del sistema.

Figura 62

Tanque de alimentación de sistema de riego



Nota: *Imagen de autoría propia.*

.

4.1.4 Implementación de componentes eléctricos

Los componentes eléctricos del sistema comprenden tanto el área de alimentación general como el circuito de control de potencia y la alimentación de la parte electrónica. A continuación, se describe cada uno de estos subsistemas y su proceso de implementación.

4.1.4.1 Protección eléctrica

Como en toda instalación eléctrica, el sistema IoT requiere una protección adecuada. Dado que el sistema se divide en dos subsistemas, se implementaron dos niveles de protección independientes. Para el circuito de potencia, considerando un consumo promedio de 12 A y los picos de corriente que ocurren durante el arranque de la bomba, se dimensionó un breaker de 30 A. Para la parte electrónica, cuyo consumo aproximado es de 6 A, se seleccionó un breaker del menor rango disponible en el mercado, acorde al consumo del subsistema electrónico.

Se muestran en la **Figura 63** la ubicación de los respectivos brekers.

Figura 63

Protección del sistema



Nota: *Imagen de autoría propia.*

4.1.4.2 Control eléctrico

Después de la protección, se implementó el circuito de control de la bomba utilizando el contactor C1 D2510, capaz de soportar picos de arranque superiores a los requeridos. La interfaz con el sistema electrónico se realiza a través de los terminales A1 y A2 de la bobina, los cuales se conectan al relé activado por la ESP32-D. Los terminales de salida T1 y T2 derivan la alimentación hacia la bomba, completando el circuito de potencia, tal como se observa en la **Figura 64**.

Figura 64

Implementación del Contactor



Nota: *Imagen de autoría propia.*

4.1.4.3 Conexión eléctrica de la bomba

La alimentación de la bomba proveniente del contactor está compuesta por tres líneas, como se observa en la **Figura 65**: fase a 110 V, neutro y tierra física. Esta última se incluye como medida de seguridad para derivar posibles sobrecargas o corrientes de fuga, protegiendo tanto al equipo como a las personas. Dicha configuración garantiza una implementación eléctrica segura, ética y conforme a las normativas vigentes para instalaciones de potencia en sistemas automatizados.

Figura 65

Conexión de la Bomba de riego



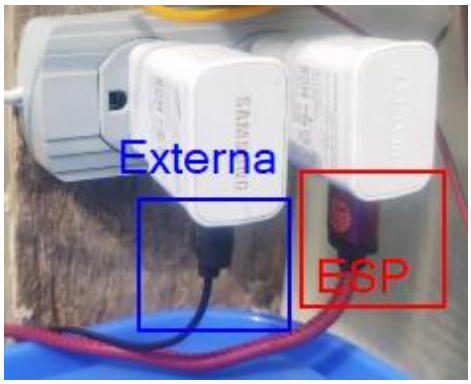
Nota: *Imagen de autoría propia.*

4.1.4.4 Alimentación de Nodo

El nodo requiere una alimentación estable de 5 V para su correcto funcionamiento. Para ello, se derivó una línea desde el breaker de protección y se instaló un tomacorriente, donde se conectan dos cargadores marca Samsung. Uno de estos cargadores alimenta la placa ESP32, mientras que el otro suministra energía a los sensores que operan a 5 V, ya que los sensores de temperatura y luminosidad son alimentados directamente por la ESP32-D. Esta configuración se ilustra en la **Figura 66**.

Figura 66

Alimentación para el Nodo y Sensores



Nota: *Imagen de autoría propia.*

4.1.5 Implementación se componentes electrónicos

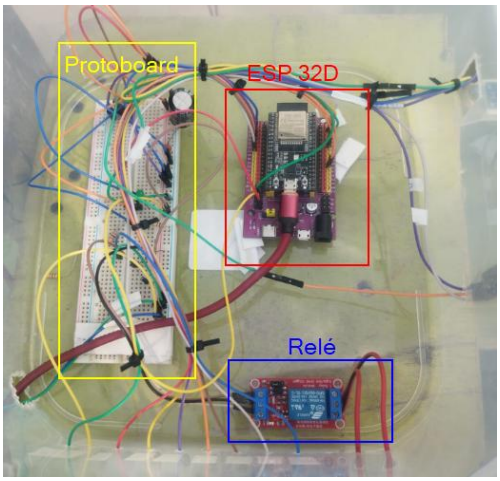
Los componentes electrónicos constituyen la base que otorga lógica y autonomía al sistema. En la presente sección se describe la ubicación y disposición física de cada uno de los sensores y placas que conforman esta arquitectura, incluyendo la caja del nodo sensor, los sensores externos y aquellos ubicados estratégicamente para la captura de todas las variables previstas en el prototipo.

4.1.5.1 Implementación caja del nodo sensor

Continuando con la disposición de los componentes, a pocos centímetros debajo de los cargadores de alimentación se encuentra la caja que aloja el protoboard, la placa ESP32-D y el relé, todos ellos dispuestos en su interior, como se observa en la **Figura 67**. El cableado interno se encuentra organizado estratégicamente para garantizar conexiones firmes y confiables entre los distintos elementos. Asimismo, la distribución de los cables que emergen al exterior ha sido planificada para facilitar el acceso y manipulación en caso de requerir mantenimiento o ajustes.

Figura 67

Caja del Nodo de sensado y actuación

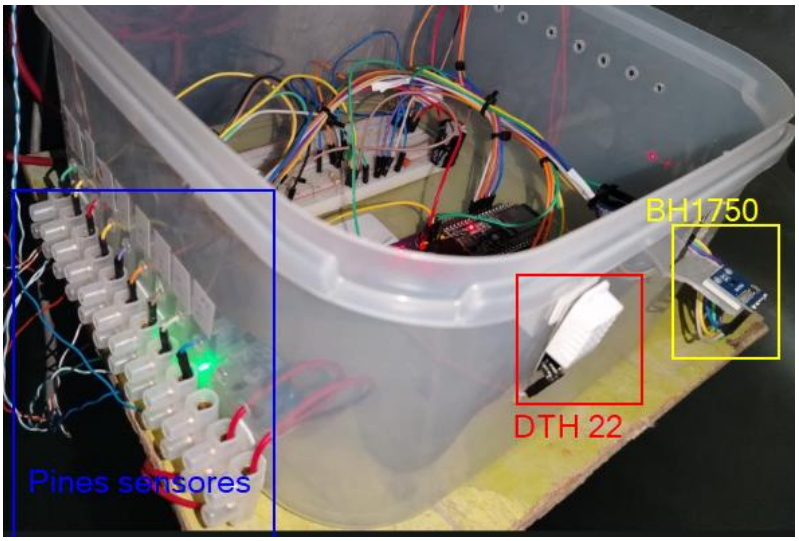


Nota: *Diseño de autoría propia.*

En la **Figura 68** se observa la parte exterior de la caja del nodo, la cual cuenta con una bornera que permite conectar y desconectar fácilmente la caja de los sensores fijos ubicados en la estructura de riego y en el tanque de agua del sistema FVH, facilitando tareas de mantenimiento o reubicación. En los laterales de la caja se encuentran instalados el sensor de temperatura ambiental (DHT22) y el sensor de luminosidad (BH1750). Esta disposición responde a la necesidad de que dichos sensores capturen datos lo más cercanos posible a las condiciones reales que experimentarán las diferentes bandejas del cultivo de Forraje Verde Hidropónico.

Figura 68

Sensores y conexiones externas de la caja del Nodo



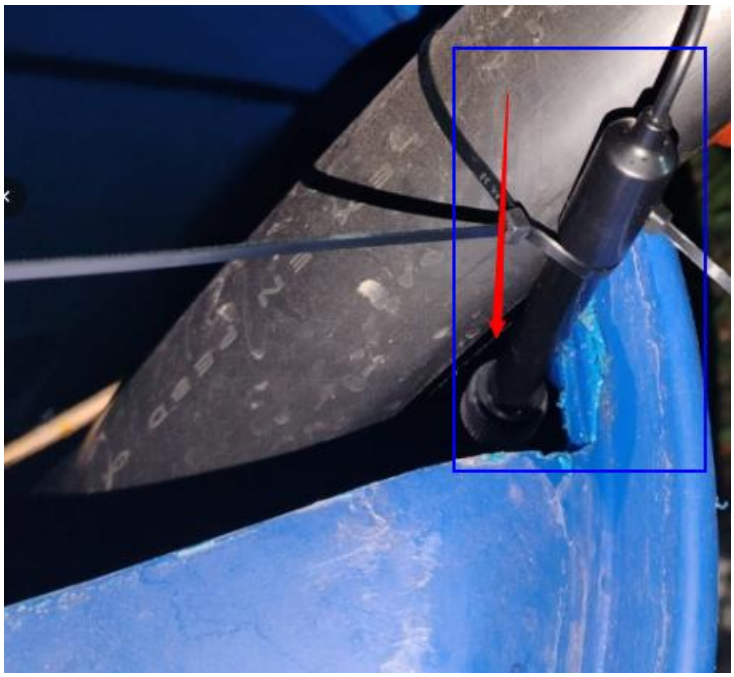
Nota: *Diseño de autoría propia.*

4.1.5.2 Implementación sensor de pH

El sensor de pH se ubica directamente en el tanque de agua que alimenta la bomba del sistema de riego. Su instalación se realizó de manera que el bulbo de medición se mantenga permanentemente hidratado, garantizando así lecturas precisas y continuas. Para asegurar su posición, el sensor fue sujetado con una amarra plástica a la manguera de succión, como se evidencia en la **Figura 69**, evitando movimientos que puedan afectar la estabilidad de la medición o dañar el componente.

Figura 69

Ubicación del sensor de pH



Nota: *Diseño de autoría propia.*

4.1.5.3 Implementación sensores de humedad

Para garantizar un monitoreo representativo de la humedad en el sistema, se establecieron puntos estratégicos de sensado en cada nivel de los estantes donde se desarrollará el FVH. Esta disposición permite obtener mediciones que reflejen el estado general de humedad en todas las áreas del cultivo, facilitando un control uniforme del riego.

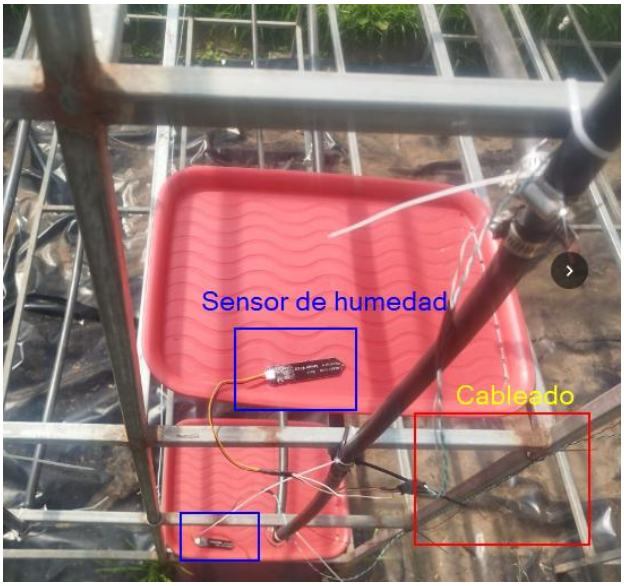
En los estantes se distribuyeron 5 puntos de sensado estratégicamente ubicados para cubrir las diferentes zonas de producción. Cabe destacar que, aunque se dispone de más de 10 sensores disponibles, estos se encuentran adheridos a las bandejas de cultivo para permitir su reemplazo individual sin afectar la continuidad del monitoreo. Sin embargo, los puntos de medición efectivos para la variable de humedad se mantienen en 5 ubicaciones fijas dentro

de la estructura.

En la **Figura 70** se ilustra la disposición final de cada uno de estos sensores dentro del sistema de estantes.

Figura 70

Ubicación de los sensores de humedad



Nota: *Diseño de autoría propia.*

4.1.6 Comprobación de sensado del sistema implementado

Una vez completada la implementación física, se procedió a evaluar el funcionamiento correcto de cada sensor de manera individual. Utilizando la comunicación serial proporcionada por la placa ESP32, se verificó el envío y recepción de datos correspondientes a cada variable monitoreada: humedad (en cada uno de los 5 puntos de sensado), temperatura, luminosidad y pH. Esta validación permitió confirmar la integridad de las lecturas y la correcta transmisión desde cada sensor hacia el nodo.

En este mismo proceso además de la obtención de datos se procedió a calibrar los sensores para que así se garantiza una correcta lectura acorde a las características reales que presenta el invernadero.

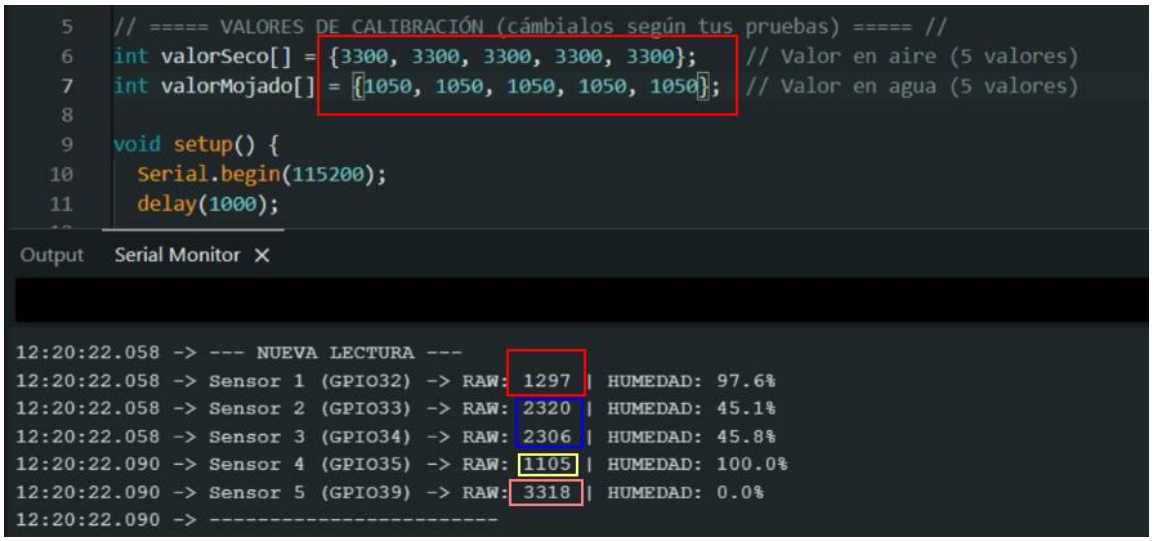
4.1.6.1 Obtención de datos sensor de humedad

Como se observa en la **Figura 71**, los sensores de humedad conectados a los pines GPIO 32, GPIO 33, GPIO 34, GPIO 35 y GPIO 39 fueron expuestos a diferentes condiciones de humedad para establecer su curva de calibración. Los sensores en los pines GPIO 32 y GPIO 35, expuestos a niveles altos de humedad, arrojaron valores analógicos cercanos a 1100 mediante la función analogRead del entorno Arduino. A partir de estas lecturas, se estableció que valores de 1050 o superiores representan el 100% de humedad.

Para niveles medios de humedad, los sensores conectados a GPIO 33 y GPIO 34 presentaron lecturas aproximadas de 2300, lo que permitió asociar este valor a un 45% de humedad. Finalmente, el sensor en GPIO 35, expuesto a condiciones de sequedad total, registró un valor de 3318, estableciéndose así que lecturas de 3300 o superiores corresponden al 0% de humedad. Esta caracterización permite mapear las lecturas analógicas a porcentajes de humedad de las cuales se calcula el promedio para su posterior envío al Edge Gateway.

Figura 71

Obtención de datos en los sensores de humedad



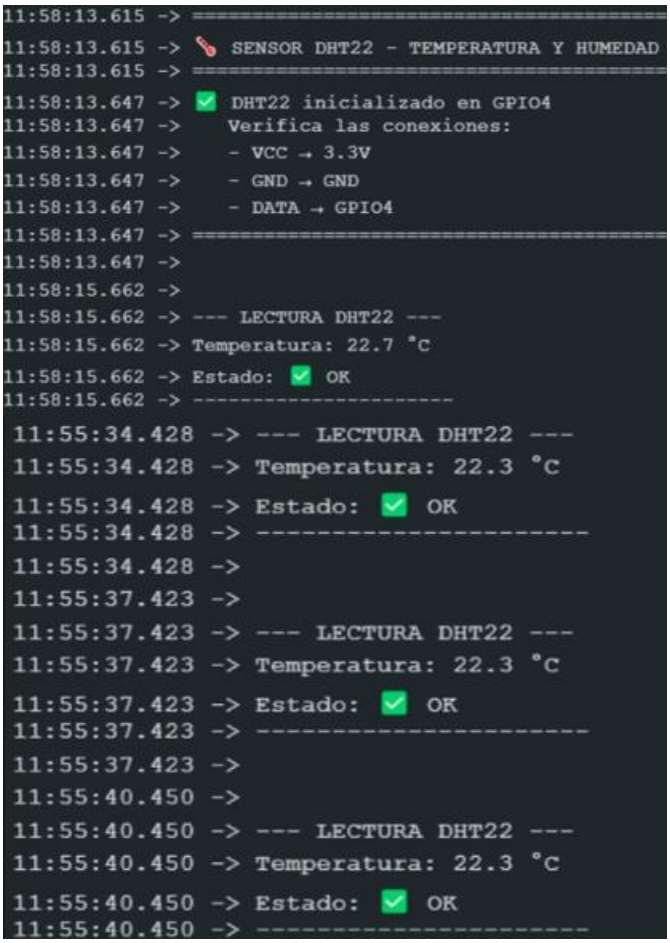
Nota: *Diseño de autoría propia.*

4.1.6.2 Obtención de datos sensor de temperatura

Como se observa en la **Figura 72**, el sensor de temperatura DHT22 opera correctamente alimentado a 3.3 V, voltaje proporcionado directamente por la placa ESP32-D. Gracias a su protocolo de comunicación digital single-wire, el sensor transmite los datos de temperatura y humedad al pin GPIO 04 del microcontrolador. Durante su funcionamiento normal, las lecturas obtenidas pueden visualizarse a través del monitor serie del entorno Arduino, lo que permite verificar en tiempo real la adquisición correcta de los datos ambientales.

Figura 72

Funcionamiento del Sensor de Temperatura



Nota: *Diseño de autoría propia.*

4.1.6.3 Obtención de datos sensor de luz

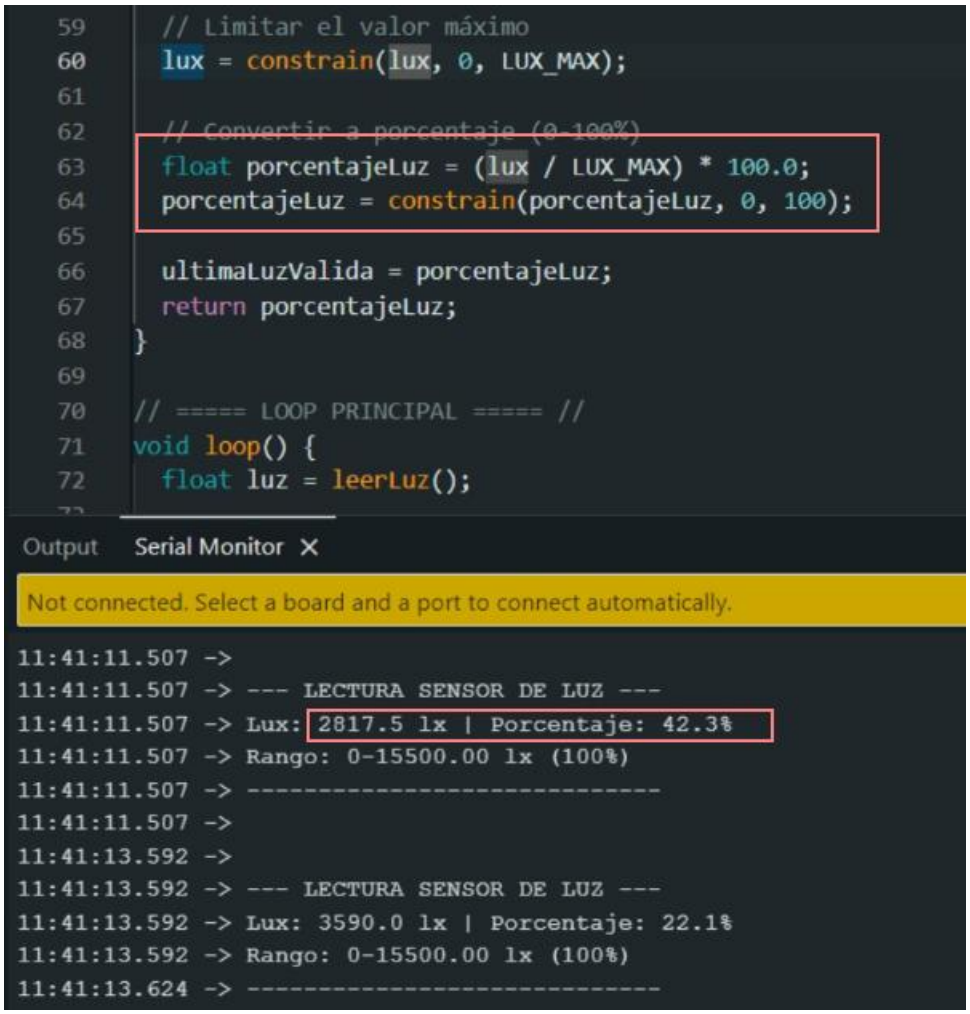
El sensor de luz BH1750 realiza la medición de los niveles de luminosidad mediante la detección de la cantidad de lux provenientes de la radiación solar. Los datos obtenidos son transmitidos a través del protocolo de comunicación I2C, utilizando los pines SDA (GPIO 22) y SCL (GPIO 21) de la placa ESP32-D. Durante su funcionamiento normal, el sensor entrega los valores digitales al microcontrolador a través del pin GPIO 22, los cuales

pueden visualizarse mediante el monitor serie de la **Figura 73**.

Para la correcta interpretación de las lecturas, se establecieron previamente los límites de calibración: el valor máximo de 15500 lux corresponde a la medición realizada bajo exposición directa al sol, mientras que el valor mínimo se estableció en 0 lux para condiciones de oscuridad total. Estos rangos permiten ponderar adecuadamente las lecturas obtenidas durante el monitoreo continuo del sistema.

Figura 73

Funcionamiento del Sensor de Luminosidad BH1750



Nota: *Diseño de autoría propia.*

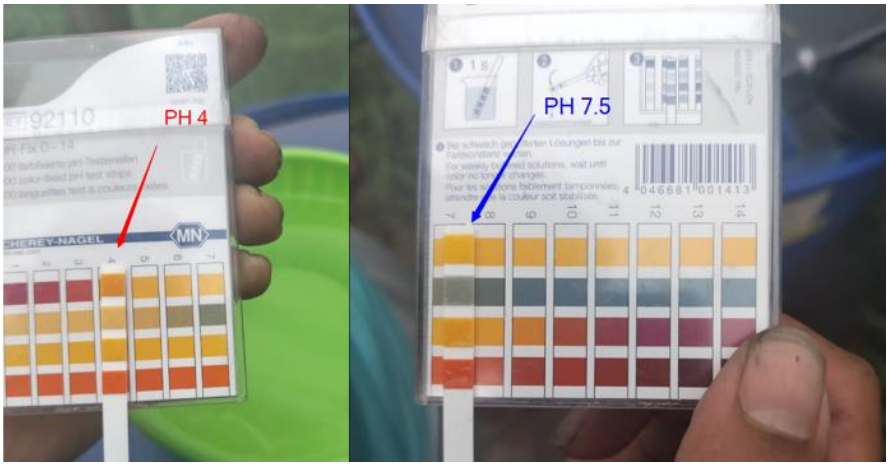
4.1.6.4 Obtención de datos sensor de pH

El sensor de pH envía la señal analógico-digital al pin GPIO 36 de la placa ESP32-D. Debido a que este sensor requiere un tiempo de estabilización para obtener lecturas confiables, el proceso de medición inicia con una etapa de estabilización antes de proceder al monitoreo continuo.

Para la calibración y verificación de su correcto funcionamiento, se sumergió la sonda en dos soluciones con valores de pH conocidos: 4.0 y 7.5, utilizando como referencia cintas medidoras de pH para confirmar los valores de las soluciones patrón. Este procedimiento, documentado en la **Figura 74**, permitió ajustar los parámetros de conversión del sensor y garantizar la precisión de las futuras mediciones en el sistema.

Figura 74

Medición de pH con cinta medidora



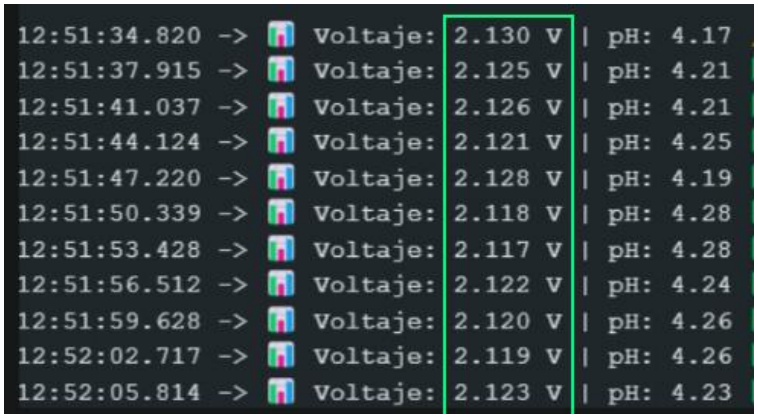
Nota: *Imagen de autoría propia.*

Mediante la comunicación serial, se observó que el voltaje medido en la solución de pH 4 fue aproximadamente de 2.15 V, mientras que para la solución de pH 7.5 se registró un valor cercano a 1.71 V. Con estos datos, se implementaron los parámetros de calibración en el

código del sensor, permitiendo convertir correctamente las lecturas de voltaje a valores de pH. La verificación final de las lecturas calibradas puede observarse a través de la comunicación serial, tal como se documenta en las **Figuras 75 y 76**.

Figura 75

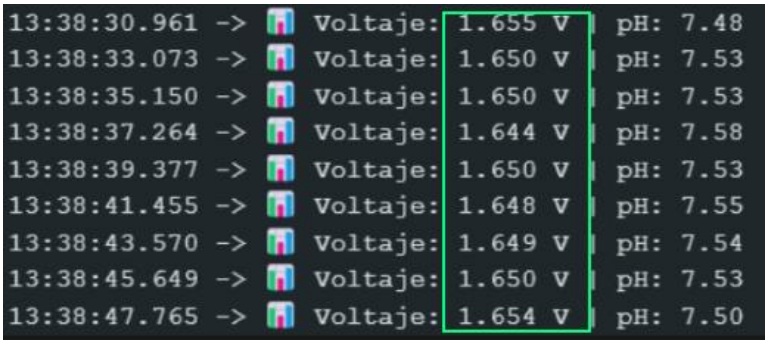
Funcionamiento del Sensor en pH 4



Nota: *Diseño de autoría propia.*

Figura 76

Funcionamiento del Sensor en pH 7.5



Nota: *Diseño de autoría propia.*

4.1.7 Análisis del proceso de comunicación del sistema

El presente capítulo describe y analiza el proceso de comunicación integral del sistema IoT, desde la adquisición de datos mediante los sensores hasta la interpretación de la información por parte del usuario final a través de notificaciones de mensajería, dashboard y aplicación de usuario. El objetivo es desglosar técnicamente el flujo de comunicación y el funcionamiento de cada componente involucrado.

Este análisis se realiza mediante el estudio de cada una de las capas del sistema IoT, empleando herramientas de monitoreo de red como TShark y Wireshark para la captura y análisis de tramas, permitiendo comprender el comportamiento de cada proceso de comunicación en las diferentes etapas del sistema.

4.1.7.1 Capa de Dispositivos

En esta capa, los sensores toman las distintas variables del ambiente y las convierten en señales analógico-digitales. Estas señales son recibidas por los pines GPIO del microcontrolador ESP32-D, cuya verificación de recepción de datos fue previamente documentada en el punto 4.1.6. En la **Tabla 32** se desglosa la composición de los datos y el proceso de comunicación correspondiente a cada sensor.

Tabla 32

Comunicación de sensores hacia al esp 32

Fase	**Origen**	**Destino**	**Tecnologías / Protocolos**	**Identificadores / Rutas**	**Puertos**
Sensado	Sensores	ESP 32D	I2C/ADC/ single-wire	mqtt_topic_sensores	Físicos GPIO

Nota: *Diseño de autoría propia.*

4.1.7.2 Capa de Red

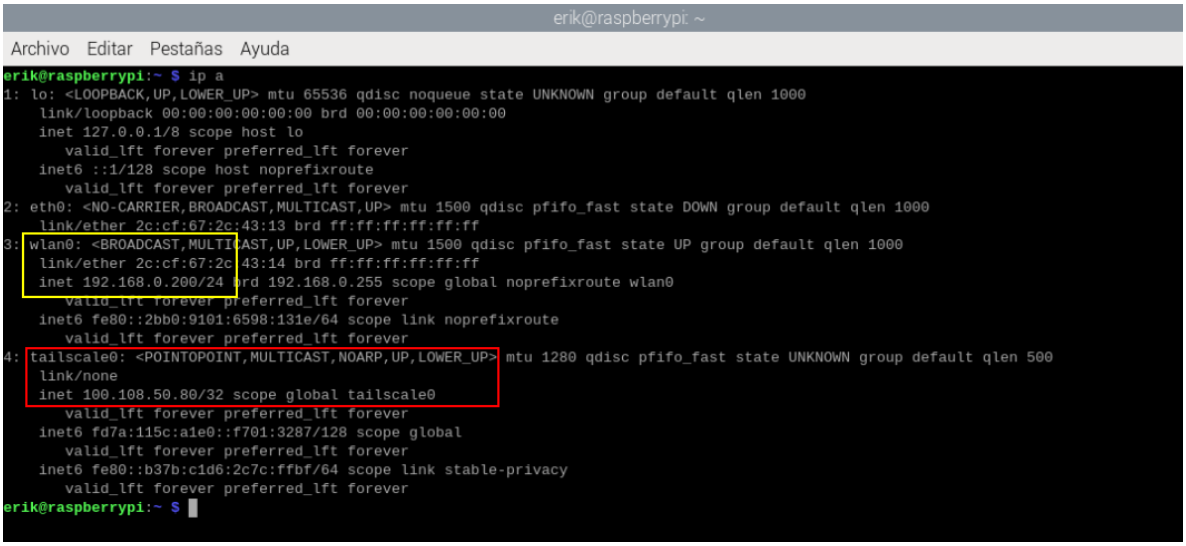
En la capa de red se desarrollan múltiples fases de negociación, suscripción y publicación de datos mediante diferentes protocolos y relaciones entre dispositivos, con el objetivo de permitir la visualización de la información en la interfaz gráfica. A continuación, se describen cada uno de los pasos realizados en este proceso.

4.1.7.2.1 Conexión red local

En esta etapa interviene la conexión inalámbrica entre el microcontrolador y la placa Edge Gateway a través de la red Wi-Fi, utilizando como intermediario el router TP-Link EX225. Para garantizar la comunicación estable entre ambos dispositivos, se asignó una IP fija al Gateway dentro del rango de direcciones proporcionado por el router, como se observa en la **Figura 77**. Adicionalmente, se observa una interfaz extra correspondiente a Tailscale, una VPN que permite acceder a los dispositivos de la red local desde cualquier ubicación externa.

Figura 77

Interfaces de comunicación del Edge Gateway

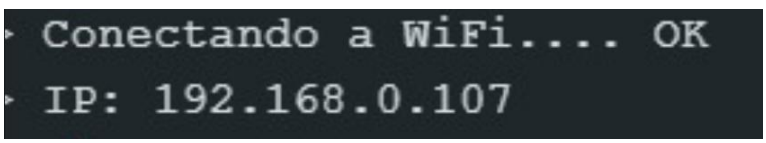


Nota: *Diseño de autoría propia.*

En el caso del microcontrolador, el código implementado en Arduino realiza la solicitud de conexión mediante el ingreso del SSID y la contraseña de la red Wi-Fi. A través del monitor serie, se puede verificar la dirección IP asignada por el servidor DHCP del router, siendo la dirección 192.168.0.107, la cual queda registrada en la lista de usuarios del dispositivo. Esta asignación se confirma mediante la instrucción *Serial.println(WiFi.localIP());* como se evidencia en la **Figura 78**.

Figura 78

Conexión de la placa ESP32-D a la red Wi-Fi



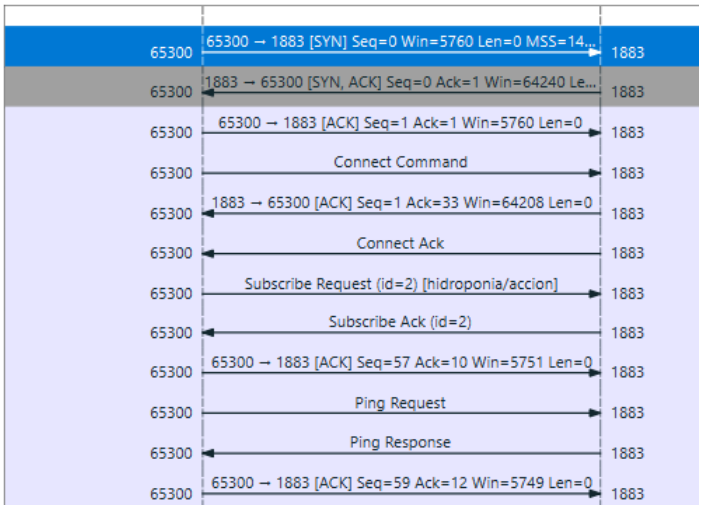
Nota: *Diseño de autoría propia.*

4.1.7.2.2 Conexión y comunicación MQTT

Una vez establecida la conectividad entre dispositivos, se inicia el proceso de negociación entre el nodo y el Edge Gateway. El nodo envía una solicitud SYN (synchronize) para iniciar el handshake TCP, a lo cual el Gateway responde con SYN-ACK (synchronize-acknowledge). Completada esta fase de negociación, el nodo procede a enviar el comando Connect para establecer la sesión MQTT, recibiendo la confirmación correspondiente por parte del Gateway. Acto seguido, el nodo solicita la suscripción a los tópicos necesarios para poder enviar y recibir mensajes. Una vez finalizada la conexión y suscripción, ambos dispositivos mantienen un mecanismo de control típico del protocolo TCP para verificar que el nodo permanece activo. Los puertos de comunicación utilizados en este proceso son el 65300 por parte del ESP32-D y el puerto estándar MQTT 1883. Este flujo de conexión puede observarse en la gráfica de flujo obtenida mediante Wireshark, presentada en la **Figura 79**.

Figura 79

Flujo de conexión y suscripción MQTT



Nota: *Diseño de autoría propia.*

Una vez establecida la comunicación, el nodo ejecuta el proceso de sensado en los intervalos programados. Tras adquirir los datos, construye un objeto JSON que encapsula todas las variables medidas. El mensaje payload es publicado utilizando el tópico sensor/data. Simultáneamente, el nodo se mantiene a la escucha de instrucciones de activación o desactivación del riego a través del tópico *hidroponia/accion* y reporta el estado del sistema mediante el tópico hidroponía/estado. Estos identificadores se encuentran definidos en el código fuente del nodo, como se ilustra en la **Figura 80**.

Figura 80

JSON elaborado por el nodo para envío al Gateway

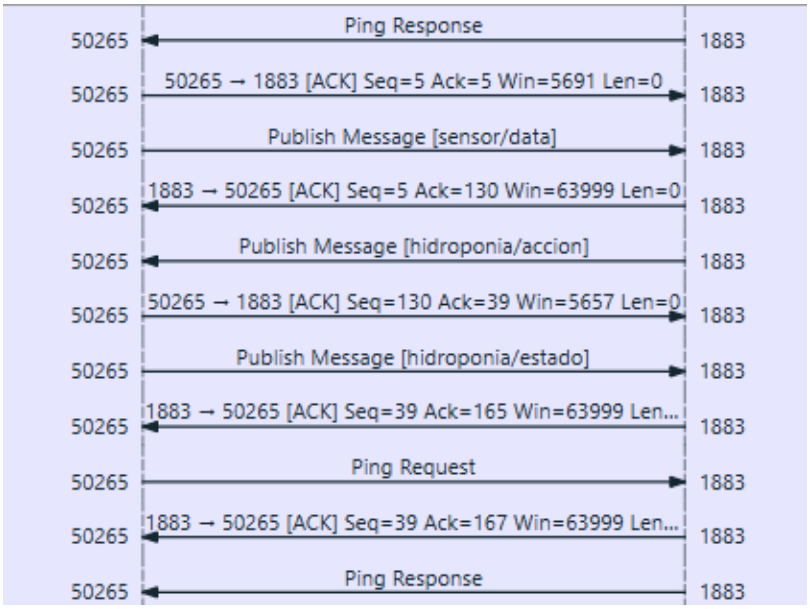


Nota: *Diseño de autoría propia.*

Mediante la herramienta TShark, instalada en la placa Raspberry Pi 5, se realizó la captura y análisis de las tramas generadas durante el proceso de comunicación MQTT, identificando los puertos 1883 y 50265 utilizados en cada uno de los extremos. En la **Figura 81** se observan los mensajes Publish intercambiados entre nodo y Gateway, cada uno con su respectivo identificador de tópico.

Figura 81

Comunicación mediante MQTT



Nota: *Diseño de autoría propia.*

Lo anteriormente descrito se resume en la **Tabla 33**, elaborada a partir de las capturas de tramas realizadas:

Tabla 33

Comunicación Nodo y Gateway

Fase	**Origen**	**Destino**	**Tecnologías / Protocolos**	**Identificadores / Rutas**	**Puertos**
Asociación Wi-Fi	ESP 32D: 192.168.0.107	Pi 5: 192.168.0.200	802.11n	SSID: Flia Cuasquer	Canal: 3-20MHz
Conexión	ESP 32D:	Pi 5:	802.11n/ tcp	Connect/Sucribe	1883/65300

MQTT Comunicación MQTT	192.168.0.107 ESP 32D: 192.168.0.107	192.168.0.200 Pi 5: 192.168.0.200	802.11n/ mqtt/tcp	hidroponia/estado Topic: hidroponia/accion Topic: sensor/data	1883/50265

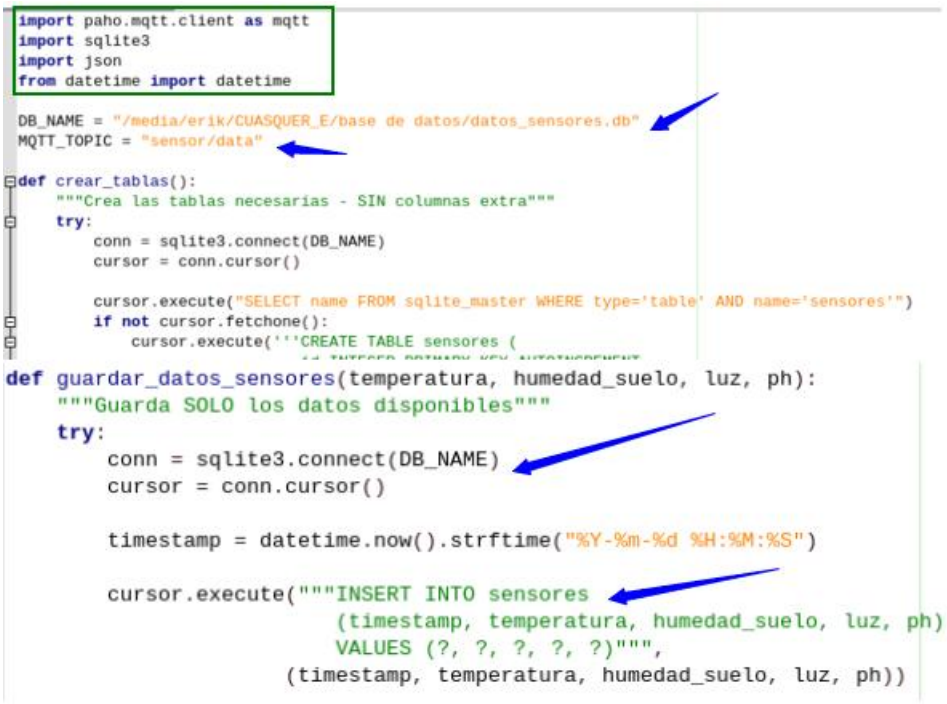
Nota: *Diseño de autoría propia.*

4.1.7.3 Capa de Apoyo

En esta sección se describe el funcionamiento del Edge Gateway, implementado sobre una Raspberry Pi 5 con dirección IP 192.168.0.200, el cual lleva a cabo las tareas de procesamiento y almacenamiento de la telemetría proveniente de los nodos sensores. La recepción de los datos se realiza a través del protocolo MQTT, utilizando la librería paho.mqtt.client para la implementación del cliente suscriptor. El flujo de procesamiento inicia con la llegada de un mensaje al tópico sensor/data, cuyo payload es descomprimido y decodificado mediante la instrucción msg.payload.decode(). Posteriormente, la cadena JSON resultante es transformada en un objeto manipulable a través de json.loads(payload), lo que permite extraer cada una de las variables medidas por el nodo. Una vez estructurados, los datos son persistidos en el sistema gestor de base de datos SQLite3, ejecutando las sentencias de inserción correspondientes sobre la tabla sensores de la base de datos datos_sensores.db, alojada en la ruta /media/erik/CUASQUER_E/. Este proceso de escritura se ilustra en la **Figura 82**.

Figura 82

Guardado de datos en SQLite



```
import paho.mqtt.client as mqtt
import sqlite3
import json
from datetime import datetime

DB_NAME = "/media/erik/CUASQUER_E/base de datos/datos_sensores.db"
MQTT_TOPIC = "sensor/data"

def crear_tablas():
    """Crea las tablas necesarias - SIN columnas extra"""
    try:
        conn = sqlite3.connect(DB_NAME)
        cursor = conn.cursor()

        cursor.execute("SELECT name FROM sqlite_master WHERE type='table' AND name='sensores'")
        if not cursor.fetchone():
            cursor.execute('''CREATE TABLE sensores (

def guardar_datos_sensores(temperatura, humedad_suelo, luz, ph):
    """Guarda SOLO los datos disponibles"""
    try:
        conn = sqlite3.connect(DB_NAME)
        cursor = conn.cursor()

        timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")

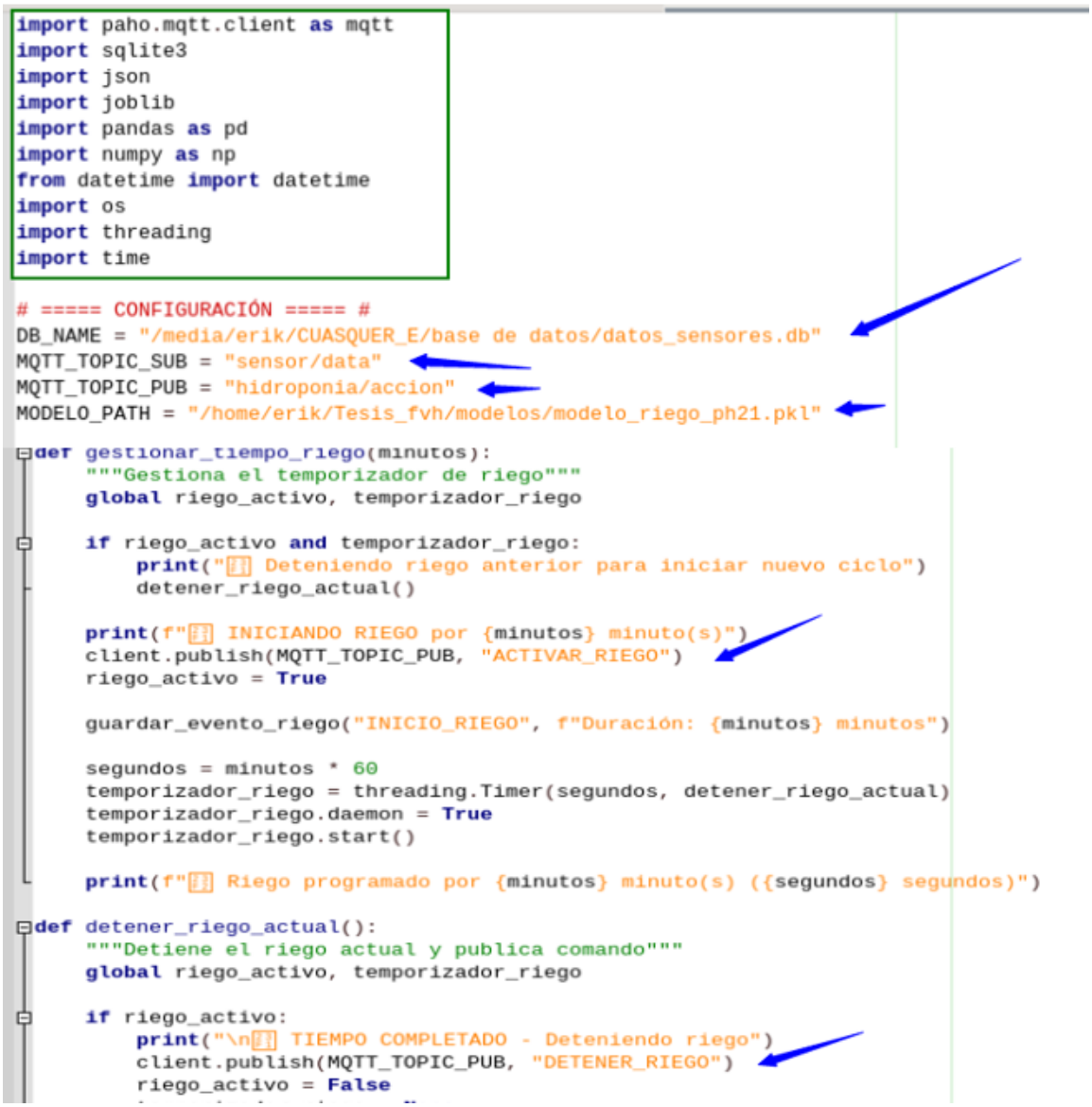
        cursor.execute("""INSERT INTO sensores
                          (timestamp, temperatura, humedad_suelo, luz, ph)
                          VALUES (?, ?, ?, ?, ?)""",
                       (timestamp, temperatura, humedad_suelo, luz, ph))
```

Nota: *Diseño de autoría propia.*

De manera paralela, el sistema de inferencia opera también de forma local en el mismo equipo. Dicho sistema, implementado en el script inferencia_riego.py, se suscribe a los mismos flujos de datos MQTT. Al recibir una nueva muestra, el sistema evalúa las variables utilizando un modelo de clasificación basado en árboles de decisión, previamente entrenado y serializado en el archivo /Tesis_fvh/modelos/modelo_riego_ph21.pkl. Como resultado de esta inferencia, se determina la necesidad de activar o desactivar el sistema de riego. La acción resultante es publicada en el tópico hidroponia/accion, lo que desencadena el comando correspondiente sobre la bomba de agua. Este ciclo de control, desde la recepción de los datos hasta la activación del actuador, se representa en la **Figura 83**.

Figura 83

Control de activación o desactivación de riego



```
import paho.mqtt.client as mqtt
import sqlite3
import json
import joblib
import pandas as pd
import numpy as np
from datetime import datetime
import os
import threading
import time

# ===== CONFIGURACIÓN ===== #
DB_NAME = "/media/erik/CUASQUER_E/base de datos/datos_sensores.db"
MQTT_TOPIC_SUB = "sensor/data"
MQTT_TOPIC_PUB = "hidroponia/accion"
MODELO_PATH = "/home/erik/Tesis_fvh/modelos/modelo_riego_ph21.pkl"

def gestionar_tiempo_riego(minutos):
    """Gestiona el temporizador de riego"""
    global riego_activo, temporizador_riego

    if riego_activo and temporizador_riego:
        print("[] Deteniendo riego anterior para iniciar nuevo ciclo")
        detener_riego_actual()

    print(f"[] INICIANDO RIEGO por {minutos} minuto(s)")
    client.publish(MQTT_TOPIC_PUB, "ACTIVAR_RIEGO")
    riego_activo = True

    guardar_evento_riego("INICIO_RIEGO", f"Duración: {minutos} minutos")

    segundos = minutos * 60
    temporizador_riego = threading.Timer(segundos, detener_riego_actual)
    temporizador_riego.daemon = True
    temporizador_riego.start()

    print(f"[] Riego programado por {minutos} minuto(s) ({segundos} segundos)")

def detener_riego_actual():
    """Detiene el riego actual y publica comando"""
    global riego_activo, temporizador_riego

    if riego_activo:
        print("\n[] TIEMPO COMPLETADO - Deteniendo riego")
        client.publish(MQTT_TOPIC_PUB, "DETENER_RIEGO")
        riego_activo = False
```

Nota: *Diseño de autoría propia.*

Con lo anteriormente expuesto, se describe que tanto el sistema de inferencia como la comunicación operan de manera local en el Edge Gateway, utilizando la ruta de la base de datos previamente definida. Dado que las interacciones se realizan mediante llamadas

locales, los puertos de comunicación externos no aplican en este nivel; en su lugar, la **Tabla 34** detalla los componentes, tecnologías y rutas involucradas en la comunicación interna de esta capa.

Tabla 34

Capa apoyo Comunicación

Fase	**Origen**	**Destino**	**Tecnologías / Protocolos**	**Identificadores / Rutas**	**Puertos**
Script de Inferencia	Pi 5: 192.168 .0.200	localhost	Python 3.10 Paho-MQTT Joblib 1.2 Pandas 2.0 NumPy	Script: inferencia_riego.py Modelo:/Tesis_fvh/modelos/modelo_riego_ph21.pkl	1883 (MQTT) File I/O
Script Suscriptor MQTT	Pi 5: 192.168 .0.200	localhost	Python 3.10 Paho-MQTT SQLite3	Script: suscriptor_mqtt.py Topics: sensor/data	1883 (MQTT) File I/O
Base de Datos SQLite	Pi 5: 192.168 .0.200	/media/erik/ CUASQUE R_E/	SQLite 3.40	DB: datos_sensores.db Tabla: sensores Tabla: eventos_riego	File I/O

Nota: *Diseño de autoría propia.*

4.1.7.4 Capa de Aplicación

En la capa de aplicación, con el objetivo de habilitar la comunicación externa, se emplean los servicios de Tailscale, una red privada virtual (VPN) que asigna direcciones IP a los dispositivos del sistema, permitiendo el acceso desde cualquier ubicación geográfica. A continuación, se describe el proceso de comunicación entre el cliente y el servidor de la placa principal, así como la integración con servicios externos como Telegram.

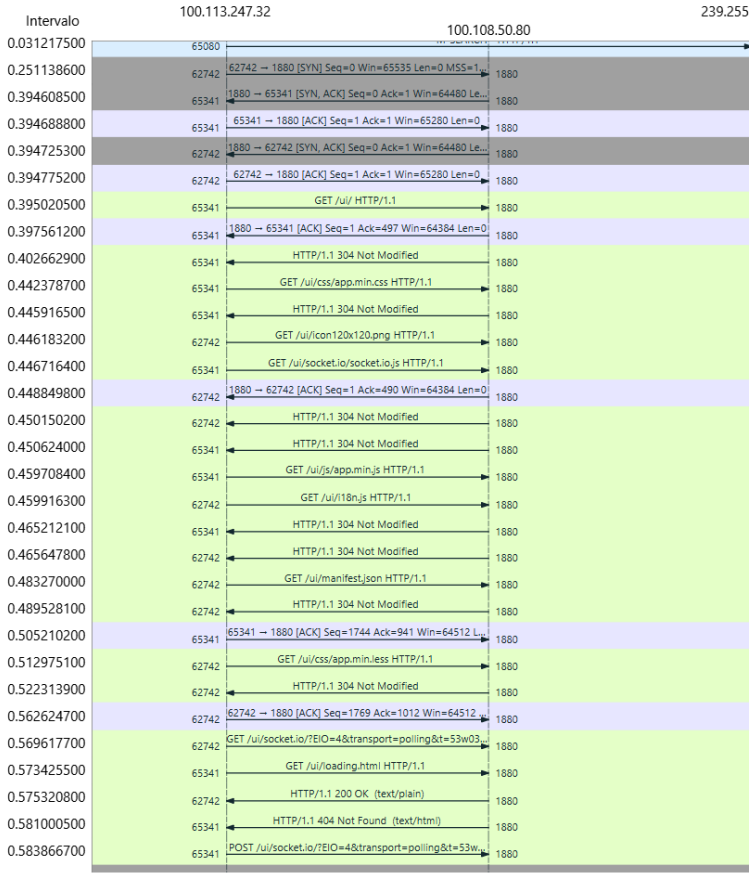
4.1.7.4.1 Acceso Remoto al Panel de Control Node-RED

El análisis del tráfico de red evidencia que el acceso al panel de control de Node-RED se realiza a través de la red virtual provista por Tailscale, utilizando inicialmente el protocolo TCP como mecanismo de transporte. Se observa el establecimiento de la conexión mediante el tradicional three-way handshake (SYN, SYN/ACK, ACK) entre el cliente (100.113.247.32) y el servidor (100.108.50.80), empleando el puerto 1880, característico del servicio web de Node-RED. Posteriormente, el cliente realiza múltiples solicitudes HTTP (método GET) orientadas a la carga de recursos del dashboard, tales como archivos CSS, JavaScript y configuraciones del entorno. Las respuestas del servidor incluyen códigos 200 OK y 304 Not Modified, lo que indica la implementación de mecanismos de caché para optimizar la transferencia de datos y reducir la latencia en la carga de la interfaz. Una vez cargados los recursos iniciales, el sistema implementa un esquema de comunicación en tiempo real mediante Socket.IO, comenzando con una fase de long polling sobre HTTP, donde se intercambian solicitudes GET y POST hacia el endpoint /socket.io. Este mecanismo actúa como transición hacia una comunicación más eficiente, concretándose posteriormente en una actualización del protocolo (upgrade) a WebSocket, evidenciada por la respuesta 101 Switching Protocols. A partir de este punto, se establece

un canal bidireccional persistente que permite el intercambio continuo de datos en tiempo real, incluyendo la transmisión de información de sensores, la actualización dinámica del dashboard y la ejecución de comandos por parte del usuario. Adicionalmente, se identifican peticiones a la API interna, como /api/login, utilizadas para la autenticación del sistema, consolidando así un flujo de comunicación estructurado, eficiente y orientado a aplicaciones IoT. Este flujo de comunicación se presenta en el diagrama de la **Figura 84**.

Figura 84

Diagrama de flujo de comunicación en la capa de aplicación.



Nota: *Diseño de autoría propia.*

4.1.7.4.2 Integración con Telegram

La comunicación con Telegram sigue un modelo cliente-servidor donde el bot implementado en Node-RED actúa como cliente y los servidores de Telegram como el servicio remoto. El proceso inicia cuando ocurre un evento en el sistema, como una decisión de riego, lo que activa una función en Node-RED que prepara un mensaje con el contenido de la alerta. Este mensaje se envía mediante una conexión HTTPS saliente desde la Raspberry Pi (192.168.0.200) hacia los servidores de Telegram en api.telegram.org a través del puerto 443, estándar para tráfico web seguro. La solicitud incluye el token único del bot (obtenido de @BotFather) para autenticación y el chat ID del destinatario (usuario o grupo) para enrutamiento. Telegram procesa la petición y entrega el mensaje al cliente correspondiente (aplicación móvil, web o desktop) en tiempo real.

En la **Tabla 35** se presenta el desglose detallado de cada uno de los componentes, tecnologías, identificadores y puertos involucrados en la comunicación de la capa de aplicación.

Tabla 35

Comunicación capa aplicación

Fase	Origen	Destino	Tecnologías / Protocolos	Identificadores / Rutas	Puertos
Node-RED Dashboard	Pi 5: 192.168.0.200 Tailscale: 100.108.50.80	- PC: 192.168.0.103 - Tailscale: 100.113.247.32 - Cualquier navegador LAN	HTTP/1.1 WebSocket Chart.js Node-RED Red Mesh VPN	URL LAN: http://192.168.0.200:1880/ui URL Tailscale: http://100.108.50.80:1880/ui	1880 41641 (UDP - WireGuard) 1880 (Node-RED) 1883 (MQTT)
Node-RED Flows	Pi 5: 192.168.0.200	Node-RED runtime	JSON JavaScript	Archivo: ~/.node-red/flows.json Nodos: ui_form, ui_chart, ui_table, ui_template, sqlite, mqtt	File I/O
Históricos	Node-RED Pi5:192.168.0.200	SQLite DB USB storage	SQLite3	SELECT * FROM sensores WHERE timestamp	File I/O
Telegram	Pi 5: 192.168.0.200	api.telegram.org	HTTPS node-red-contrib-telegrambot	Botname: riegofvhbot Chat id: -1003465361503 Token: 8146515091: AAFRvvzyfSdkBj	443

Nota: *Diseño de autoría propia.*

4.1.8 Diagrama de comunicación final del sistema

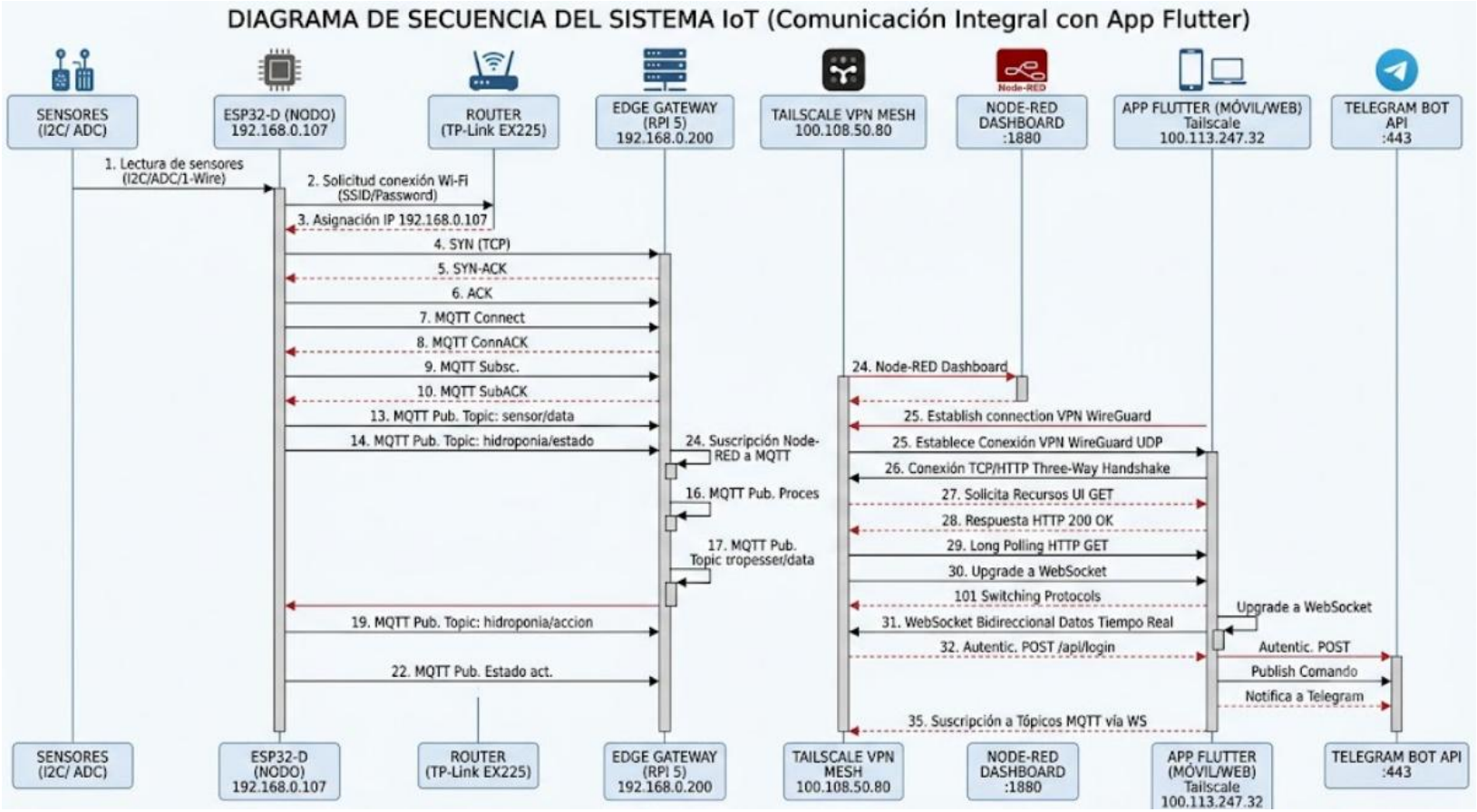
Tras realizar las capturas y el análisis de la comunicación de cada una de las partes del sistema, se presenta en la **Figura 85** el diagrama general del proceso de comunicación, en el cual se muestran las solicitudes, respuestas y enlaces entre los diferentes componentes que integran y estructuran el sistema IoT.

Como se observa en la figura, el flujo de comunicación involucra los siguientes elementos:

- Sensores y nodo de sensado (ESP32): encargados de la adquisición de variables ambientales.
- Canal de comunicación inalámbrica: representado por el router y el protocolo IEEE 802.11.
- Edge Gateway (Raspberry Pi 5): responsable del procesamiento, almacenamiento y lógica de control.
- VPN (Tailscale): utilizada para la conexión remota segura con los dispositivos finales.
- Sistema de monitoreo (Node-RED): plataforma de visualización y control en tiempo real.
- Dispositivos móviles: acceso a través de la aplicación desarrollada en Flutter.
- API de Telegram: para el envío de alertas y notificaciones periódicas al usuario.

Este diagrama permite visualizar de manera integral la interacción entre cada una de las capas del sistema, desde la captura de datos hasta la entrega de información al usuario final.

Figura 85
Diagrama de Comunicación



Nota: *Diseño de autoría propia*

4.1.9 Comprobación de monitoreo del sistema

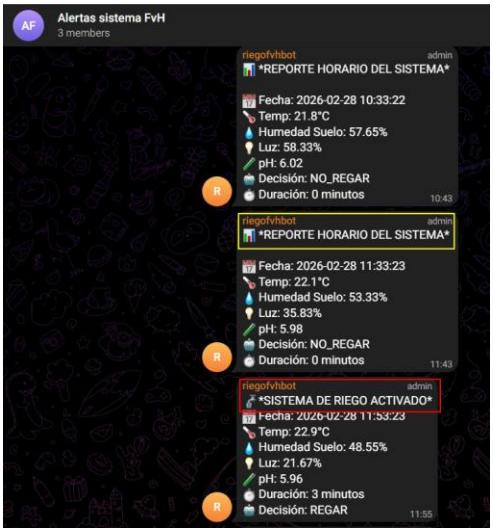
En esta sección revisamos las comprobaciones del funcionamiento de lo que el usuario final va a observar, como son las notificaciones enviadas mediante telegram y los gráficos de monitoreo, así como el historial de registros actual del sistema y los distintos gráficos de análisis a los que podemos acceder en el dashboard.

4.1.9.1 Envío de mensajes telegram

Los mensajes de alerta enviados a través de Telegram constituyen una parte fundamental para la comprobación del correcto funcionamiento del sistema. Para ello, se verificaron las notificaciones generadas por el sistema una vez transcurrido el intervalo programado para los reportes horarios, así como los avisos emitidos en momentos especiales, como la activación del sistema de riego. En la **Figura 86** se evidencia la recepción de dichas notificaciones, confirmando que el sistema de alertas opera con normalidad.

Figura 86

Envío de alertas de Telegram



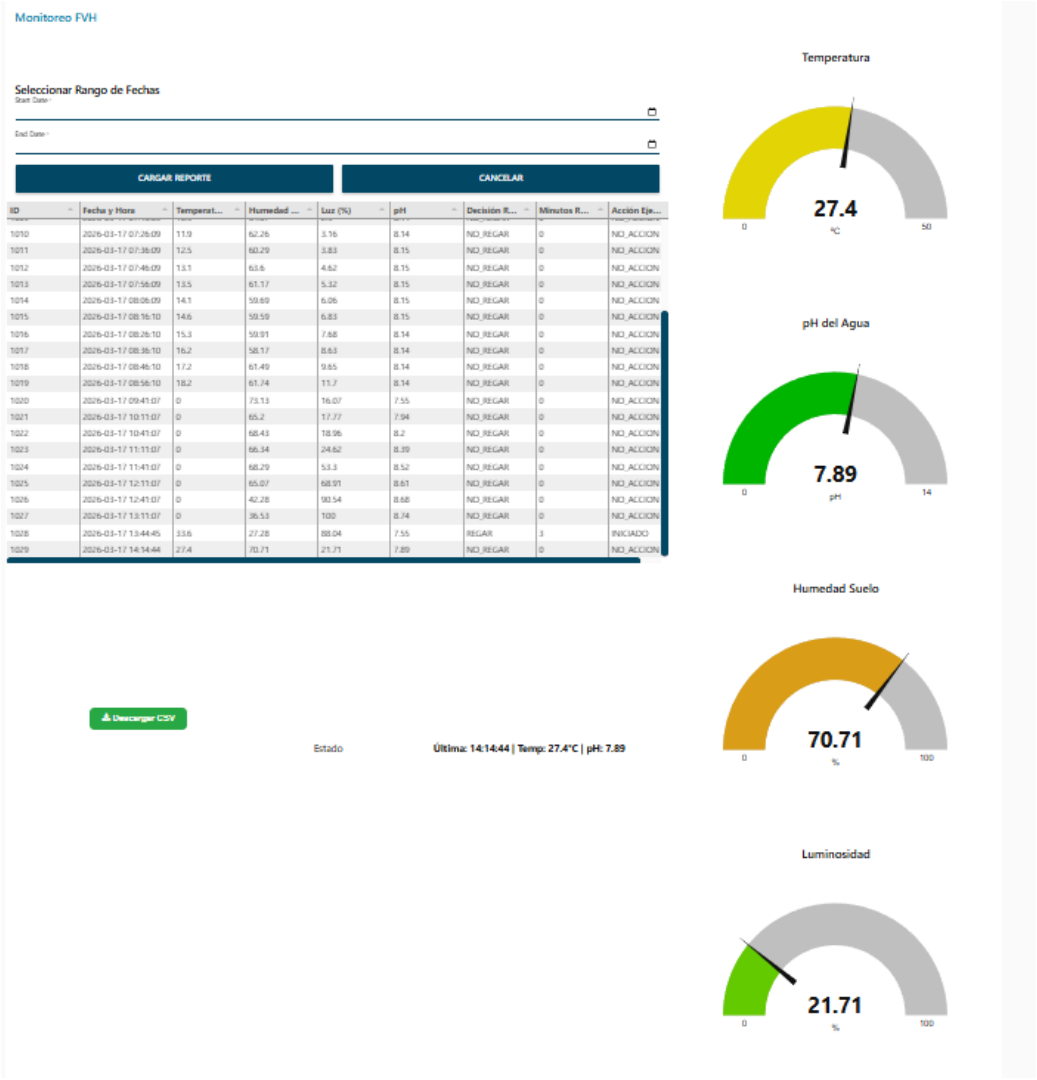
Nota: *Diseño de autoría propia.*

4.1.9.2 Comprobación monitoreo en dashboard de node-red

Mediante la función del dashboard podemos observar que existe una buena comunicación entre el dashboard con cada una de las partes del sistema ya que los datos que está recibiendo son reales y actualizados de acuerdo a los tiempos previstos como se evidencia en la **Figura 87**.

Figura 87

Dasboard funcional



Nota: *Diseño de autoría propia.*

4.2 Pruebas

Durante la etapa de pruebas, se verificó el correcto funcionamiento de cada componente del sistema, validando la precisión de los sensores, la estabilidad de la comunicación inalámbrica entre nodos y gateway, y la eficiencia del sistema de aspersión de agua. Además, se realizaron pruebas de estrés para garantizar que el sistema operara de manera óptima bajo diferentes condiciones ambientales, asegurando su confiabilidad antes de la puesta en marcha definitiva.

4.2.1 Planificación del cultivo FVH

Para dimensionar la capacidad operativa del sistema, se determinaron las características y el potencial productivo esperado. Esto incluyó el cálculo del número total de bandejas que pueden ser acomodadas en los estantes, la cantidad de bandejas que deben procesarse diariamente para mantener una producción continua, así como los tiempos específicos de germinación y desarrollo de las plantas de forraje verde hidropónico (FVH). Este análisis permite establecer un flujo de trabajo sistematizado que optimiza el espacio, los recursos y el ciclo de cultivo, en donde tomamos en cuenta los siguientes datos:

Datos base de tu sistema

- Total, de bandejas: **96**
- Ciclo FVH trigo: **15 días**
- Riegos: **depende del nivel promedio de humedad**
- Humedad objetivo (según histórico):
  55–65 % (ideal ≈ 60 %)
- Producción deseada: **continua, diaria**
- Estantes: siempre **completos**

Con los parámetros establecidos, se generó la ecuación para determinar el número de bandejas que deben sembrarse y cosecharse diariamente. Considerando un ciclo completo de cultivo de 15 días, el cálculo se realiza mediante la fórmula:

$$\frac{96 \text{ Bandejas}}{15\ Dias} = 6.4 \approx 6\text{–}7\ bandejas \qquad \ldots(11)$$

De este resultado se deduce que, en una semana de 7 días, se deben sembrar 6 bandejas diarias durante los primeros seis días, y una séptima bandeja en el último día para compensar la diferencia decimal. Esta planificación asegura una producción continua y evita que los estantes queden vacíos, manteniendo un flujo constante de FVH.

Sin embargo, para el arranque inicial del sistema, es necesario llenar por completo las 96 bandejas durante la primera semana. Para ello se calcula:

$$\frac{96 \text{ bandejas}}{5 \text{ días}} = 19.2 \qquad \ldots(12)$$

Este ritmo de siembra intensiva durante los primeros cinco días permite alcanzar la capacidad total del sistema, dando inicio al ciclo productivo continuo que se describe en la primera **Tabla 36** de implementación.

Tabla 36

Planificación de cultivo inicial

Día	Bandejas sembradas	Acumulado	Etapa dominante	Humedad objetivo	Riego
1	20	20	Germinación	65–70 %	3 riegos
2	20	40	Germinación	65–70 %	3 riegos
3	19	59	Germinación / crecimiento	60–65 %	3 riegos
4	19	78	Crecimiento	58–62 %	3 riegos
5	18	96	Crecimiento	55–60 %	3 riegos

Fuente: Autoría

Esta estrategia de arranque permite establecer una tolerancia operativa dentro de los márgenes productivos esperados del sistema. Una vez completada la fase inicial de llenado, se procede a implementar el cronograma de producción constante basado en los cálculos previamente establecidos.

Tabla de producción constante

A partir de la segunda semana, el sistema opera bajo el cronograma de siembra y cosecha diaria mostrado en la **Tabla 37**, diseñado para mantener un flujo continuo y optimizar el uso de la capacidad instalada:

Tabla 37

Planificación constante del sistema

Día del ciclo	Acción	Bandejas	Humedad objetivo	Riego
6	Sembrar	6	58–62 %	3
7	Sembrar	6	58–62 %	2–3
8	Sembrar	6	55–60 %	3
9	Sembrar	6	55–60 %	2–3
10	Sembrar	6	55–60 %	2–3
11	Sembrar	6	52–58 %	2–3
12	Sembrar	6	52–58 %	2
13	Sembrar	6	50–55 %	2–3
14	Sembrar	6	50–55 %	2
15	Sembrar	6	50–55 %	2–3

Nota: *Diseño de autoría propia.*

Es decir, se espera mantener una siembra y cosecha diaria de estos forrajes como se nota en la **Tabla 38**.

Tabla 38

Calculo cosecha esperada del sistema

Todos los días	Acción
Cosechar	6 bandejas (15 días de edad)
Sembrar	6 bandejas nuevas
Total, en estantes	Siempre 96
Humedad promedio	≈ 60 %

Nota: *Diseño de autoría propia.*

Con este diseño se busca mantener un sistema de riego automatizado que garantice una producción constante de forraje de trigo, asegurando su desarrollo óptimo y saludable. El control preciso de la humedad ambiental y del sustrato previene la proliferación de hongos y evita deficiencias en el crecimiento, manteniendo las condiciones ideales para cada etapa del ciclo del FVH mediante una humedad promedio regulada y estable.

4.2.2 Primera etapa de pruebas

En esta sección se busca verificar la funcionalidad del sistema para el monitoreo de datos reales una vez iniciado el proceso de siembra del Forraje Verde Hidropónico (FVH). Bajo la guía del ingeniero agrónomo, se realizó el procedimiento completo para la implementación de las bandejas, abarcando las etapas de desinfección, germinación, desarrollo y consumo del producto.

4.2.2.1 Preparación de las semillas

De acuerdo con las indicaciones del ingeniero Jairo Hernández, se realizó el proceso de

preparación de las semillas para su pre-germinación. Con el fin de prevenir la formación de hongos nocivos durante la producción de forraje, se aplicó un tratamiento con hidróxido de calcio (cal apagada) en una concentración de 50 gramos de cal por litro de agua, como se observa en la **Figura 88**.

Figura 88

Preparación para germinación en agua con cal



Nota: *Imagen de autoría propia*

.

En esta fase también se realizó la selección y eliminación de semillas defectuosas, tal como se muestra en la **Figura 89**, con el objetivo de evitar que el cultivo se deteriore por causa de estas. Este proceso permitió obtener únicamente las semillas aptas para la germinación, las cuales recibieron el tratamiento de pre-germinación para aprovechar el agua y reducir el tiempo del proceso. El tiempo de remojo fue de 8 horas, seguido de 1 hora de oxigenación de la semilla, para luego repetir el remojo por 8 horas adicionales.

Figura 89

Selección de semillas aptas para proceso

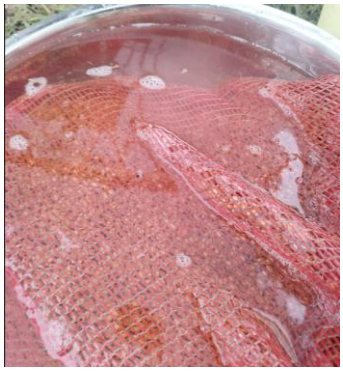


Nota: *Imagen de autoría propia.*

Posteriormente, se procedió a la etapa de oreado, donde las semillas fueron colocadas en un saquillo de color rojo que proporciona una adecuada ventilación. Durante esta etapa se produce una liberación de calor considerable en el grano, lo que estimula el crecimiento del embrión, como se evidencia en la **Figura 90.**

Figura 90

Oreado de las semillas



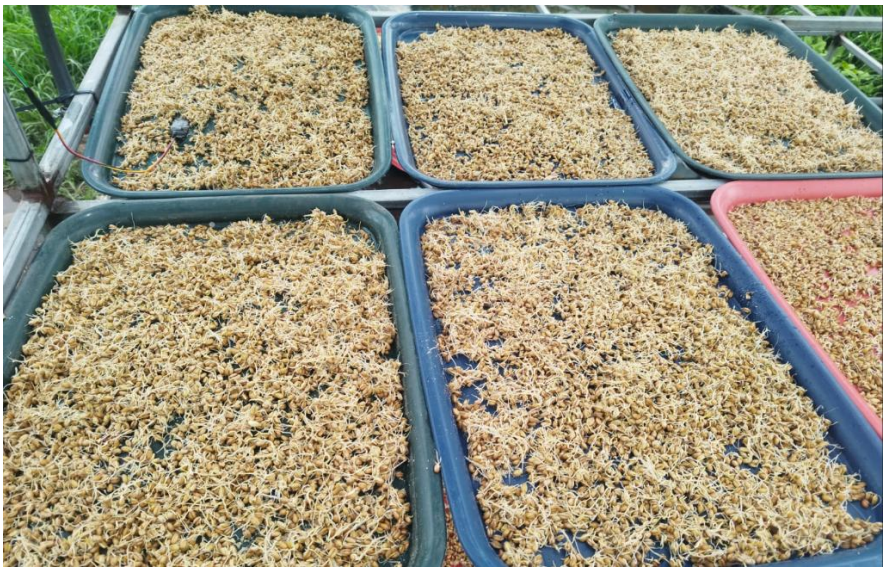
Nota: *Diseño de autoría propia.*

4.2.2.2 Preparación de bandejas en los estantes

Una vez completada la etapa de pre-germinación, se procedió a la distribución homogénea de las semillas en las bandejas de cultivo, las cuales fueron dispuestas en los estantes metálicos siguiendo un diseño en batería para optimizar el espacio vertical, tal como se muestra en la **Figura 91**. Paralelamente, se realizó la ubicación de sensores de humedad dentro de las bandejas designadas, con el fin de monitorear en tiempo real la capacidad de campo y la disponibilidad hídrica para las plántulas, permitiendo así ajustes precisos en la frecuencia y duración del riego.

Figura 91

Ubicación Bandejas en estantes



Nota: *Imagen de autoría propia.*

4.2.2.3 Desarrollo de las semillas

A los tres días posteriores a la siembra, se constató que el sistema automatizado de riego operó en función de la humedad remanente del sustrato y de las variables microclimáticas

del ambiente, garantizando condiciones óptimas para la germinación y el crecimiento inicial. Se registró una humedad relativa promedio del 60%, valor considerado dentro del rango más adecuado para la producción de FVH, favoreciendo la actividad enzimática y la elongación celular. En la **Figura 92** se observaron las primeras radículas y coleóptilos emergiendo, evidenciando un inicio homogéneo del desarrollo vegetativo.

Figura 92

Bandejas al tercer día de sembradas



Nota: *Imagen de autoría propia.*

Se evidenció un ritmo de crecimiento constante, con incrementos morfológicos significativos cada 48 horas, acordes a las curvas de crecimiento esperadas para el cultivo de trigo hidropónico. Los brotes alcanzaron una altura comprendida entre 0.5 y 1 cm, como se evidencia en la **Figura 93**, presentando una coloración verde uniforme y turgencia adecuada, indicadores de un adecuado estado fitosanitario y nutricional.

Figura 93

Medición de brotes de trigo



Nota: *Imagen de autoría propia.*

Entre el cuarto y quinto día después de la germinación, las plántulas desarrollaron un sistema radicular denso y enmarañado formando una estera radicular, así como un tallo de entre 6 y 7 cm de altura, tal como se aprecia en la **Figura 94**.

Figura 94

Crecimiento de primeros días del fvh



Nota: *Diseño de autoría propia.*

4.2.2.4 Consumo del producto final

Cuando el forraje verde hidropónico alcanzó una altura de cosecha de 16 a 20 cm, mostrado en la **Figura 95**, se determinó el punto óptimo de madurez fisiológica para su uso como suplemento alimenticio. En este estado, el FVH presentó una alta relación hoja:tallo y una frondosidad característica, lo que lo convierte en un alimento de alto valor nutricional y proteico para la crianza de cuyes (Cavia porcellus).

Figura 95

FVH listo para la cosecha.



Nota: *Imagen de autoría propia.*

Dicho forraje fue ofrecido directamente a los animales en el criadero designado, como se registra en la **Figura 96**, quienes lo consumieron de manera eficiente, demostrando su aceptabilidad y su potencial como complemento forrajero en sistemas de producción agropecuaria.

Figura 96

Consumo del producto final de fvh.



Nota: *Imagen de autoría propia.*

4.3 Pruebas de Hardware/Software

En esta sección del capítulo cuatro se presenta el cronograma de pruebas realizadas para validar el funcionamiento del sistema IoT aplicado al cultivo de forraje verde hidropónico (FVH) de trigo. Las pruebas fueron ejecutadas de manera progresiva, considerando la integración de hardware y software, así como la comunicación entre los diferentes módulos del sistema. En la **Tabla 39** se presenta de forma detallada las actividades realizadas, los componentes involucrados, el entorno de prueba, los resultados esperados y el tiempo designado para su ejecución.

Tabla 39

Cronograma de pruebas del funcionamiento del sistema

Nº de Prueba	**Detalle de la prueba**	**Involucrados**	**Hardware/Software**	**Lugar de pruebas**	**Resultados a esperar**	**Duración**
1	Verificación del funcionamiento de sensores (humedad, DHT22, pH y luminosidad)	Autor	ESP32, sensores	Área de cultivo FVH	Lecturas estables, coherentes y sin ruido	01 al 03 de abril
2	Prueba de acoplamiento físico del nodo de adquisición	Autor	ESP32, protoboard/circuito	Nodo de sensores	Componentes correctamente conectados y protegidos	04 al 06 de abril
3	Envío de datos desde ESP32 hacia Edge gateway	Autor	ESP32, Raspberry pi	Red local	Datos transmitidos sin pérdida	07 al 10 de abril

4	Almacenamiento de datos en base de datos SQLite	Autor	SQLite, Node-RED	Servidor local	Datos almacenados con ID y timestamp	11 al 14 de abril
5	Integración y validación del modelo de IA (árbol de decisión)	Autor	Python	Servidor de procesamiento	Clasificación correcta del estado del cultivo	15 al 18 de abril
6	Visualización en dashboard de Node-RED	Autor	Node-RED Dashboard	Navegador web	Datos visibles en tiempo real	19 al 23 de abril
7	Prueba de comunicación remota mediante Tailscale	Autor	Tailscale	Red virtual	Acceso remoto seguro al sistema	24 al 27 de abril
8	Validación de la aplicación móvil y notificaciones Telegram	Autor	Flutter, Telegram, Node-RED	Dispositivo móvil	Visualización correcta, envío de alertas y control de activación.	28 al 30 de abril

Nota: *Diseño de autoría propia.*

4.3.1 Pruebas de Acoplamiento de los Componentes

Se realizaron pruebas orientadas a verificar la correcta integración física de los componentes del sistema, específicamente en el nodo de adquisición de datos. Este nodo está compuesto por el microcontrolador ESP32 y los sensores: sensor capacitivo de humedad, sensor de temperatura y humedad ambiental, sensor de pH y sensor de luminosidad, después de haber sido implementados se verifico que estén funcionales y en buen estado antes de proceder con las pruebas y calibración.

El objetivo de estas pruebas es garantizar que los dispositivos se encuentren correctamente instalados, protegidos frente a condiciones ambientales y operando dentro de rangos adecuados. En la **Tabla 40** se presenta la descripción y cumplimiento de estas.

Tabla 40

Pruebas de acoplamiento y funcionalidad partes del nodo sensor.

N°	Descripción de la prueba	Criterio de evaluación	Sí	No
1	Integración del sensor de humedad capacitivo a la ESP 32	Lecturas estables	X	
2	Integración del sensor DHT22 a la ESP	Datos coherentes	X	
3	Instalación del sensor de pH a la ESP	Lectura sin interferencias	X	
4	Ubicación del sensor de luminosidad	Medición adecuada	X	
5	Protección del circuito	Sin exposición externa	X	
6	Estabilidad del ESP32	Sin reinicios	X	
7	Precisión de sensores	Valores dentro de rangos	X	

Nota: *Diseño de autoría propia.*

Los resultados obtenidos evidencian un correcto acoplamiento de los componentes electrónicos, garantizando estabilidad en la adquisición de datos y protección frente a factores ambientales como humedad o polvo. Asimismo, se verificó la correcta ubicación de los sensores y la adecuada realización de las conexiones eléctricas, aspectos fundamentales para asegurar la seguridad operativa y la durabilidad de los componentes a lo largo del tiempo.

Desde el punto de vista agronómico, las condiciones ambientales medidas por el sistema permitieron observar un desarrollo adecuado del cultivo de Forraje Verde Hidropónico (FVH) de trigo, evidenciando que el entorno controlado cumplió con los requerimientos necesarios para el crecimiento óptimo de las plantas.

4.3.2 Pruebas de Comunicación de componentes del sistema

Las pruebas presentadas en la **Tabla 42** validan la transmisión de datos entre el nodo de adquisición (ESP32), el nodo central (Raspberry Pi 5) y los dispositivos del usuario final. La evaluación abarcó aspectos como el empaquetado de los datos sensados, la verificación del tiempo de envío y la coherencia de los datos recibidos, con el fin de comprobar la comunicación bidireccional entre los dispositivos de sensores y actuadores dentro de la arquitectura del sistema.

Tabla 41

Pruebas de comunicación entre nodos

N°	Descripción de la prueba	Criterio de evaluación	Sí	No
1	Envío de datos desde ESP32	Transmisión correcta	X	
2	Recepción en Edege gateway	Datos completos	X	
3	Integridad de datos	Sin alteraciones	X	
4	Frecuencia de envío	Intervalos constantes	X	
5	Respuesta de Edege gateway a ESP32	Envío comandos correctos	X	
6	Recepción de comandos en ESP32	Transmisión correcta	X	
7	Acceso remoto de dispositivos a Edege gateway	Conectividad estable mediante Tailscale	X	

Nota: *Diseño de autoría propia.*

Se comprobó la correcta comunicación entre los componentes del sistema, asegurando la transmisión continua de datos y el control desde el Edge Gateway y desde los dispositivos móviles de forma remota. El uso de Tailscale permitió un acceso remoto seguro, facilitando el monitoreo del sistema sin necesidad de infraestructura en la nube. Esto posibilitó correlacionar en tiempo real las variables medidas con el comportamiento del cultivo.

4.3.3 Pruebas de Funcionamiento del Nodo Central

En esta etapa se evaluó el funcionamiento del nodo central o Edge Gateway, encargado del almacenamiento, procesamiento y análisis de los datos provenientes del nodo de sensado. Este módulo integra Node-RED como plataforma de monitoreo, una base de datos SQLite, un modelo de aprendizaje automático basado en árbol de decisión desarrollado en Python, y

una aplicación móvil programada en Flutter para el monitoreo y control remoto del sistema. En la **Tabla 41** se describen las pruebas realizadas dentro del nodo central para verificar su correcto comportamiento, abarcando aspectos como la recepción y almacenamiento de datos, la ejecución del modelo de inferencia, la publicación de comandos de riego y la respuesta de la interfaz de usuario.

Tabla 42

Pruebas de nodo Edge Gateway

	Descripción de la prueba	Criterio de evaluación	Sí	No
1	Integridad de los datos recibidos	Datos completos sin errores	X	
2	Creación de base de datos SQLite	Tablas correctas	X	
3	Inserción de datos	Registros realizados correctamente	X	
4	Integración del modelo IA	Toma de decisiones correctamente	X	
5	Presentación en Node-RED	Dashboard funcional	X	
6	Envío de comandos de decisión	Decisión coherente	X	
7	Generación de alertas	Telegram, y app generan alertas predeterminadas	X	

Nota: *Diseño de autoría propia.*

Como se evidencia en la **Tabla 41**, los resultados de las pruebas realizadas al nodo central demostraron un desempeño eficiente en el manejo de datos, abarcando desde su almacenamiento hasta su procesamiento. El modelo de árbol de decisión implementado

permitió evaluar el estado del cultivo de FVH, generando resultados coherentes y acordes con las condiciones observadas en el entorno de producción.

La plataforma de visualización Node-RED facilitó un monitoreo continuo en tiempo real, brindando al usuario una interfaz clara y actualizada de las variables del sistema. Adicionalmente, se verificó la correcta vinculación entre las distintas plataformas que integran el sistema —monitoreo, control y decisión—, así como el funcionamiento adecuado de la conexión remota, garantizando así la operatividad integral del sistema IoT para la producción de FVH.

4.3.4 Pruebas de Funcionamiento de Conexión Remota con Dispositivo Móvil

En esta fase se validó la comunicación con el dispositivo móvil del usuario mediante una aplicación desarrollada en Flutter, la cual permite visualizar los datos del sistema de manera remota, apoyándose en una VPN que posibilita la interconectividad con la red local del sistema IoT. El objetivo fue confirmar la confiabilidad de la comunicación y la veracidad de los datos mostrados en la aplicación, así como verificar el funcionamiento de las alertas en Telegram y el control remoto a través del chat de dicha plataforma.

La **Tabla 43** muestra las pruebas realizadas en esta sección.

Tabla 43

Pruebas de comunicación móvil

N°	Descripción de la prueba	Criterio de evaluación	Sí	No
1	Comunicación Node-RED – App	Conexión estable	X	
2	Recepción de datos	Datos correctos	X	
3	Visualización de variables	Interfaz funcional	X	

4	Actualización en tiempo real	Sin retrasos	X
5	Visualización del monitoreo del cultivo	Visible	X
6	Acceso remoto (Tailscale)	Funcional	X
7	Notificaciones Telegram	Alertas correctas	X

Nota: *Diseño de autoría propia.*

La aplicación móvil permitió visualizar de manera eficiente las variables del sistema, facilitando el monitoreo remoto del sistema FVH. Se lograron identificar eventos relevantes como variaciones de humedad, cambios de temperatura, niveles de iluminación y alertas por condiciones críticas. Adicionalmente, la integración con Telegram posibilitó notificar al usuario en tiempo real, mejorando la capacidad de respuesta ante anomalías y garantizando así una comunicación efectiva entre el sistema IoT y el usuario final.

Dentro de las figuras presentadas a continuación, se evidencia cómo la aplicación móvil permite analizar el comportamiento del sistema de acuerdo con los registros realizados durante el período de funcionamiento del sistema mismo que se filtra por fechas estableciendo un periodo de tiempo específico. De esta manera, se muestra en la **Figura 97** la cantidad de registros realizados en dicho período, así como el promedio, las lecturas máximas y mínimas de las distintas variables (temperatura, humedad y pH), permitiendo así un análisis más profundo del estado del sistema.

Figura 97

Estadísticas de registros filtrados por fechas



Nota: *Diseño de autoría propia.*

En la **Figura 98** se presenta el registro del pH monitoreado, considerando que el rango óptimo para cultivos hidropónicos se encuentra entre 5.5 y 6.5. Cuando esta variable se encuentra por debajo o por encima de los valores establecidos, el sistema genera una alerta en la aplicación para informar al usuario, permitiendo corregir la condición mediante la disolución del químico adecuado, estabilizando así el pH y garantizando que el sistema decida correctamente si regar o no en el cultivo de FVH.

Figura 98

pH monitoreado por un periodo de tiempo

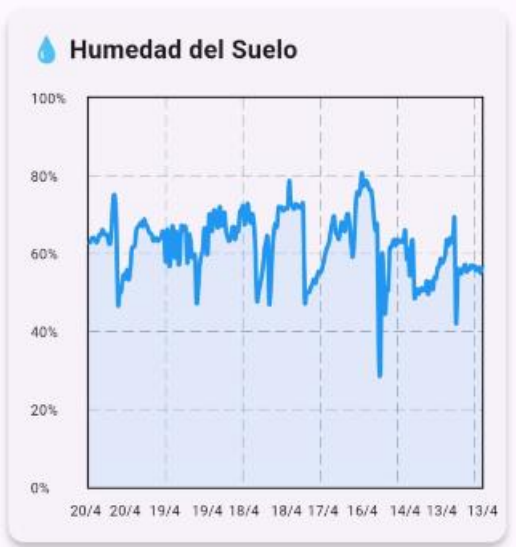


Nota: *Diseño de autoría propia.*

En cuanto a la **Figura 99**, el registro mostrado corresponde a la variación de la humedad en el sistema. Esta variable presenta mayor variabilidad, ya que depende de la temperatura registrada y de la absorción propia del cultivo de FVH. Es la variable más importante del sistema, pues con base en ella se decide si activar o no el riego. En orden de prioridad, le siguen la temperatura ambiental y el pH. La variable de luz no se muestra directamente, ya que es de menor importancia según los objetivos del sistema; sin embargo, sí es considerada en las decisiones tomadas por el árbol de decisión.

Figura 99

Humedad monitoreada por un periodo de tiempo

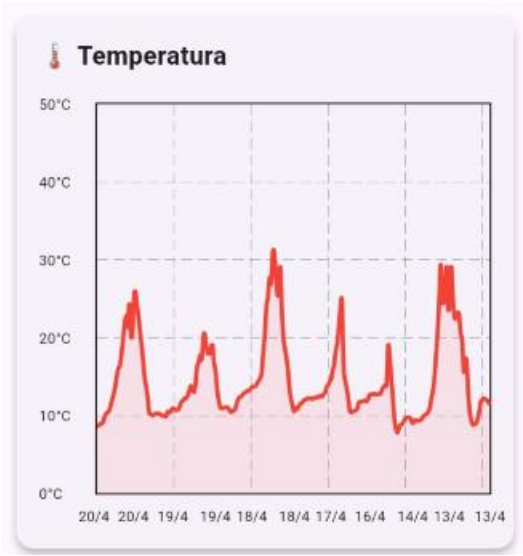


Nota: *Diseño de autoría propia.*

Finalmente, en la **Figura 100** se presenta la temperatura monitoreada. De acuerdo con la fluctuación de esta variable, depende la fluctuación que se registre de acuerdo al comportamiento del nivel de humedad del sistema. Por ello, el análisis de la temperatura permite determinar si el período de tiempo establecido puede implicar una mayor necesidad hídrica o no, contribuyendo así a que el sistema decida correctamente.

Figura 100

Temperatura monitoreada por un periodo de tiempo



Nota: *Diseño de autoría propia.*

4.4 Medición de Parámetros del Estado Nutricional Hidropónico

Se verifica el cumplimiento en su totalidad de las pruebas realizadas sobre la producción de FVH, lo que permite afirmar que el sistema IoT automatizado desempeña el objetivo deseado. En consecuencia, se realiza la toma de datos en el ambiente para determinar, según estas mediciones, el estado nutricional procesado por el algoritmo de aprendizaje supervisado "Árbol de Decisión". Si bien la literatura especializada (FAO) establece rangos óptimos para cultivos hidropónicos en general, el presente sistema fue calibrado y validado para las condiciones específicas del cultivo de trigo como Forraje Verde Hidropónico (FVH) para la crianza de cuyes en el cantón San Pedro de Huaca.

La **Tabla 44** detalla los rangos operativos reales medidos por el sistema durante las fases de germinación, crecimiento y cosecha del FVH, contrastándolos con los parámetros de calidad de referencia.

Tabla 44

Rangos de Variables Ambientales y Nutritivas Medidos por el Sistema IoT-FVH

Variable / Propiedad	Unidad	Rango de Referencia	Rango Medido por el Sistema IoT	Observación
pH del Agua	-	5.5 - 6.5	6.0 - 7.5 (Inicial) 5.8 - 6.5(Controlado)	El sistema registró un pH inicial más elevado (debido al agua de red), pero tras la calibración del sensor y el monitoreo constante, se logró estabilizar en rangos óptimos para la absorción de nutrientes del trigo.
Humedad	%	50% - 70% (Rango ideal para FVH)	40% - 85% (Rango total registrado) / Activación de riego < 55%	El algoritmo "Árbol de Decisión" utilizó esta variable como principal criterio. El sistema activó el riego al detectar valores por debajo del 50-55% y lo detuvo al superar el 65-70%, demostrando un control de alta precisión.
Temperatura Ambiente	°C	15°C - 30°C (Rango óptimo para FVH)	7°C - 38°C (Registrado en el archivo CSV)	A temperaturas >32°C y alta luminosidad, el algoritmo evitó el riego para no generar estrés hídrico en las plántulas.
Luminosidad Ambiental	%	20% a 100%	Dia 100 % Noche 0%	Con muy alta luminosidad (>80%) y alta temperatura, se evitó regar para prevenir la lixiviación de nutrientes

Nota: *Diseño de autoría propia.*

4.5 Evaluación de resultados

Una vez implementado el sistema y realizadas las pruebas de funcionamiento tanto para la estructura del sistema IoT hidropónico como para el hardware y software asociados, se inició la recolección de información del cultivo, verificando así su utilidad conforme a los objetivos planteados. El cultivo de FVH de trigo fue analizado mediante un enfoque comparativo con las expectativas de crecimiento, lo que permitió evaluar su rendimiento en aspectos como crecimiento, coloración, producción y eficiencia en el ahorro de agua.

Finalmente, los resultados obtenidos determinan la viabilidad de la técnica de cultivo supervisado para la crianza de cuyes, validando la implementación del sistema automatizado como una herramienta efectiva para la producción de forraje en condiciones controladas.

4.5.1 Análisis del desarrollo del FVH de Trigo

En esta fase se realiza el análisis del cultivo de trigo para Forraje Verde Hidropónico (FVH) sembrado en el sistema hidropónico automatizado. Se utilizaron 96 bandejas distribuidas en 4 niveles, con un ciclo de cultivo planificado de 15 días hasta la cosecha. A lo largo de este proceso, el sistema IoT proporcionó riego automático basado en las decisiones del Árbol de Decisión, el cual evaluaba constantemente la humedad del sustrato, la temperatura y la luminosidad.

Para determinar y obtener resultados del cultivo, es indispensable evaluar puntos de discusión como: altura de las plantas, coloración de las hojas, desarrollo radicular y uso del agua. Estos parámetros se comparan y evalúan frente a los obtenidos en mediciones del método tradicional de siembra de trigo en suelo.

4.5.2 Altura de la Planta

4.5.2.1 Método tradicional

En cuanto al desarrollo de las semillas mediante el método tradicional, los brotes tardaron entre 4 a 5 días en emerger en el suelo. Como se observa en la **Figura 101**, el desarrollo no es homogéneo: algunas plántulas presentan una altura de 2.5 cm, mientras que otras alcanzan un máximo de 4 cm, debido a factores del suelo que no pueden ser controlados.

Figura 101

Altura primeros días del cultivo tradicional



Nota: *Imagen de autoría propia.*

El desarrollo es evidentemente más lento en suelo y no uniforme. En las zonas donde las semillas se desarrollaron adecuadamente, a los 11 días se observan plántulas de 14 cm, lo cual es un buen desarrollo. Sin embargo, en contraste, existen plántulas que apenas alcanzan 11 cm o 12 cm. En la **Figura 102** se evidencia que algunos brotes están emergiendo recién, con alturas aproximadas de 8 a 9 cm.

Figura 102

Plántulas desarrolladas en segunda semana del método tradicional



Nota: *Imagen de autoría propia.*

Finalmente, al transcurrir el tiempo, algunas plántulas se desarrollan de manera aceptable, pero al no existir uniformidad, los datos varían significativamente. En la tercera semana de desarrollo del cultivo tradicional, se observan plantas con alturas superiores a las determinadas por el sistema FVH, pero esto se debe a que el método tradicional requiere un período de maduración total de 120 a 150 días, tiempo en el cual el sistema FVH puede alcanzar hasta 10 cosechas. En la **Figura 103** se nota que aún hay plantas que no se han desarrollado adecuadamente, sin una frondosidad abundante como en el sistema FVH, con alturas que varían entre 13 cm y 23 cm máximos.

Figura 103

Medición de plantas en la tercera semana



Nota: *Imagen de autoría propia.*

Con base en estas observaciones, se construyó la **Tabla 45**, que resume el crecimiento semanal del trigo en condiciones tradicionales.

Tabla 45

Tabla de crecimiento del método tradicional

Tiempo de cultivo	**Altura mínima (cm)**	**Altura máxima (cm)**	**Altura promedio (cm)**	**Observaciones**
Día 4 - 5	2,5	4,0	3,2	Emergencia inicial de

				plántulas
Día 11	8,0	14,0	11,0	Desarrollo heterogéneo
Día 21	13,0	23,0	18,0	Persisten diferencias de crecimiento

Nota: *Diseño de autoría propia.*

4.5.2.2 Sistema FVH

En la **Figura 93** se evidenció el inicio homogéneo del desarrollo vegetativo a los 3 días, con brotes de 0.5-1 cm. La **Figura 94** mostró un desarrollo acelerado a los 5 días, con tallos de 6-7 cm. Finalmente, la **Figura 95** documenta que, a los 15 días, el FVH de trigo alcanzó una altura de cosecha de 18 a 20 cm.

Mediante estas observaciones, se concluye que esta técnica de cultivo supervisado alcanza la madurez forrajera en el tiempo planificado (15 días), demostrando la efectividad del control ambiental automatizado. La **Tabla 45** detalla el crecimiento semanal promedio del FVH de trigo, evidenciando un desarrollo acelerado en la primera semana y una madurez óptima al final de la segunda.

Tabla 46

Crecimiento periódico del FVH de Trigo

Días	**Etapa**	**Altura**	**Observaciones**
Día 0 a 2	Siembra	0 cm	Semilla de trigo seleccionada y pregerminada.
Día 3 a 4	Germinación	0.5 - 1.0	Emergencia de radículas y coleóptilos. El sistema

		cm	mantuvo humedad estable (~60%).
Día 5 a 6	Crecimiento Inicial	6.0 - 7.0 cm	Desarrollo de tallo y primeras hojas. Sistema radicular en formación (estera).
Día 7 al 10	Crecimiento Rápido	10.0 - 14.0 cm	Elongación acelerada del tallo. Incremento de la frondosidad.
Día 15	Cosecha	18.0 - 20.0 cm	Punto óptimo de madurez. Alta relación hoja: tallo. Rico en proteína y fibra.

Nota: *Diseño de autoría propia.*

4.5.3 Coloración de Hoja del FVH de Trigo

Para el análisis de coloración del Forraje Verde Hidropónico (FVH) de trigo, se realizó una comparativa visual basada en la **Tabla 47**, el FVH de trigo se consume como planta completa (tallo, hojas y raíz), por lo que la evaluación de pigmentos se realizó sobre la totalidad de la biomasa aérea.

Las observaciones realizadas durante el ciclo de cultivo de 15 días permiten deducir que las plántulas de trigo del sistema hidropónico presentaron una coloración verde intenso y uniforme en tallos y hojas, sin bordes secos, puntas amarillentas ni decoloraciones grisáceas. Esto indica un excelente estado de salud, una alta concentración de clorofila y una eficiente actividad fotosintética.

Tabla 47

Referencia de pigmentación de la planta

Pigmento	Color	Observación en el FVH de Trigo (Plántula completa)	Interpretación para el Cultivo
Clorofila a	Azul-Verde	**Presente (Dominante).** Otorga el color verde intenso característico a tallos y hojas.	Indica una **fotosíntesis eficiente** y conversión de energía lumínica (medida por el sensor BH1750) en biomasa.
Clorofila b	Verde	**Presente.** Complementa la absorción de luz, visible en los tonos verdes más brillantes de las hojas jóvenes.	**Captación de luz** en condiciones de luminosidad variable.
Carotenos Xantofilas	Amarillo Naranja	**No visibles a simple vista.** Ausencia de tonos amarillentos en tallos y hojas maduras.	**No hubo estrés hídrico ni oxidativo** significativo. Evitando la degradación de clorofilas.
Feofitina	Gris	**No presente.** Ausencia de tonos grisáceos o apagados en el forraje.	La **clorofila no se ha degradado** por exceso de calor, pH desbalanceado o senescencia prematura. El monitoreo constante aseguró condiciones óptimas durante todo el ciclo de 15 días.

Nota: *Diseño de autoría propia.*

Con respecto a esta sección, para el método tradicional el trigo se desarrolló de acuerdo con los colores normales esperados. Sin embargo, es importante señalar que en el método tradicional únicamente se pueden aprovechar las hojas y el tallo del trigo, ya que la raíz, aunque se desarrolle adecuadamente, no puede ser utilizada para el consumo de los cuyes, que es el fin con el que se siembra.

4.5.4 Frondosidad del FVH de Trigo

4.5.4.1 Método tradicional

Con el método tradicional, se observa en la **Figura 104** que en las zonas señaladas en color rojo las semillas no se desarrollaron de manera adecuada, a diferencia de las zonas en color azul. Esto demuestra que en el método tradicional se está sujeto a condiciones variables dependiendo del suelo y de distintos factores externos que impiden el desarrollo uniforme de las plántulas de trigo.

Figura 104

Frondosidad con el método tradicional.



Nota: *Imagen de autoría propia.*

4.5.4.2 Sistema IoT FVH

En comparación con el método tradicional, en el sistema IoT de FVH las semillas se encuentran aisladas de condiciones externas adversas, lo que permite un desarrollo más óptimo. La pérdida de semillas es mínima o nula, el forraje se desarrolla de manera uniforme y con una frondosidad abundante. Además, la producción por metro cuadrado se multiplica por cuatro, ya que, en la misma área de suelo utilizada en el método tradicional, en el sistema FVH se disponen 4 niveles con igual capacidad de producción.

La frondosidad observada en el cultivo de Forraje Verde Hidropónico (FVH) de trigo, documentada en la **Figura 105**, fue abundante, densa y uniforme en todas las bandejas cosechadas a los 15 días. Esta biomasa vegetal se traduce en una mayor masa alimenticia, un mejor perfil nutricional y una mayor aceptabilidad por parte de los cuyes durante las pruebas de consumo.

Figura 105

Frondosidad de FVH



Nota: *Imagen de autoría propia.*

4.5.5 Desarrollo de Raíz

4.5.5.1 Método tradicional

En el suelo, el crecimiento radicular puede verse limitado por factores como compactación del sustrato o presencia de patógenos. El desarrollo de la plántula de trigo resulta ser muy variable: en algunos casos el desarrollo es bueno, pero en otros las plántulas no se desarrollan con normalidad, como se observó en la **Figura 104**. En cuanto al desarrollo de la raíz en el método tradicional, en el recuadro rojo de la **Figura 106** se evidencia un desarrollo limitado, y esta porción no puede ser aprovechada para el consumo animal debido a la exposición a contaminación del suelo. Por lo tanto, se puede afirmar que se pierde hasta una tercera parte de la planta. En el recuadro azul se muestra la parte aprovechable, de manera que, de una planta total de 18 cm, únicamente se pueden aprovechar 14 cm.

Figura 106

Desarrollo de la raíz método tradicional



Nota: *Diseño de autoría propia.*

4.5.5.2 Sistema IoT FVH

Para el análisis y apreciación del desarrollo de la zona radicular del FVH de trigo, el ingeniero agrónomo que supervisa el desarrollo del proyecto manifiesta que, en este tipo de sistema hidropónico por aspersión, las raíces tienen libertad de crecimiento y forman una estera radicular densa y de color blanco-crema, como se observa en la **Figura 107**, lo cual es un indicador de alta actividad metabólica y óptima absorción de nutrientes.

Este vigoroso sistema radicular es fundamental para el rápido crecimiento de la biomasa aérea la cual puede ser aprovechada para su consumo aprovechando la totalidad de la planta, demostrando la superioridad de la técnica hidropónica supervisada frente a métodos tradicionales.

Figura 107

Desarrollo de raíces del FVH



Nota: *Imagen de autoría propia.*

4.5.6 Análisis del Consumo de Agua en el Sistema Hidropónico

Para este análisis, se procesaron los registros de riego comprendidos entre el 31 de marzo y el 11 de mayo de 2026. En los días donde se registraron distintos eventos de "REGAR" (automático) o "REGAR_MANUAL" de un minuto de duración, se llevó a cabo un registro y tabulación de los eventos. El consumo de agua se calculó sobre una base de 10 litros por minuto de riego, valor aproximado medido de acuerdo con la capacidad del tanque de almacenamiento y la disminución del nivel de agua cada vez que se activa la bomba.

En la **Tabla 48** se presentan el registro del consumo de agua por semana, así como el total mensual utilizado por el sistema. Adicionalmente, se incluyen distintas observaciones relacionadas con el desarrollo del cultivo monitoreado por el sistema

Tabla 48

Consumo de agua

Semana	**Fechas**	**Cambios Positivos**	**Cambios Negativos**	**Total, Riegos**	**Consumo de Agua**	**Consumo Acumulado**
Semana 1	31/03 al 06/04	Trasplante exitoso. Sistema detecta humedad adecuada (70-85%) sin necesidad de regar en días nublados.	Plántulas en adaptación inicial. Riegos manuales nocturnos esporádicos por precaución.	8	80	80
Semana 2	07/04 al 13/04	Sistema comienza a activarse ante demanda. Temperatura ambiente ideal (18-25°C). Bajo consumo.	pH del sistema se estabiliza (6.0-6.5). Se registran algunos riegos automáticos de 1 minuto por humedad baja (<50%).	10	100	180
Semana 3	14/04 al	Crecimiento acelerado de plantas (hasta 10cm). El	Incremento de temperatura (hasta 28°C) aumenta la demanda hídrica.	8	80	260

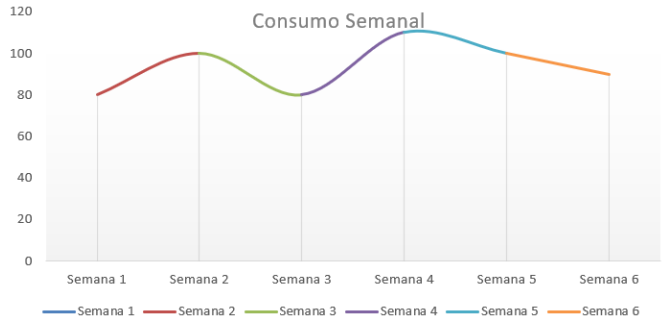
	20/04	algoritmo "Árbol de decisión" optimiza el riego.	Riegos de 3 minutos más frecuentes.			
Semana 4	21/04 al 27/04	Plantas alcanzan madurez temprana (15-18cm). Desarrollo de follaje frondoso (8-10 hojas).	Picos de temperatura (hasta 38°C) el día 26/04 disparan alarmas. Consumo alto para compensar estrés hídrico.	11	110	370
Semana 5	28/04 al 04/05	Plantas en punto óptimo de cosecha (hasta 20cm). Sistema responde bien a olas de calor.	Se detectan múltiples activaciones manuales desde App/Telegram (anomalía), se ajustan a max. 3 por día para el cálculo.	10	100	470
Semana 6	05/05 al 11/05	Ciclo completado. Consumo se estabiliza. El sistema entra en fase de mantenimiento hasta nueva cosecha.	Temperaturas nocturnas bajan (<10°C), reduciendo la evaporación y la necesidad de riego.	9	90	560

Nota: *Diseño de autoría propia.*

En la **Figura 108** se muestra gráficamente el comportamiento del consumo de agua en litros durante el período de funcionamiento y producción de FVH, permitiendo así obtener una representación visual de la eficiencia del sistema en relación con las fluctuaciones de las variables ambientales.

Figura 108

Fluctuación del consumo semanal de agua

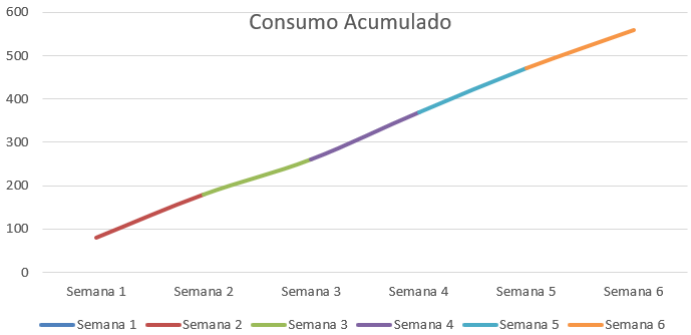


Nota: *Diseño de autoría propia.*

Adicionalmente, en la **Figura 109** se presenta la acumulación y el total de litros consumidos por el sistema durante el período de pruebas, evidenciando de manera consolidada el volumen total de agua utilizado en el proceso productivo.

Figura 109

Acumulación consumo general de agua



Nota: *Diseño de autoría propia.*

4.5.6.1 Método tradicional

Para el método tradicional realizamos el cálculo acorde al consumo estimado de acuerdo a los días en los que se realizó la siembra mediante las técnicas tradicionales en donde para cubrir un área de un 1.2 m por 1.2 aproximadamente se necesitó de entre 10 a 15 litros para cubrir esa área de riego debido a la infiltración del suelo. a la vez este riego se realiza dos veces por semana debido a que se encuentra dentro del invernadero donde se implementó el sistema, cabe recalcar que el consumo del método tradicional aumentara ya que antes de poder cosechar el producto final necesitamos esperar entre 120 a 150 días.

Datos

- Área de referencia en suelo: 1.2 m × 1.2 m = 1.44 m²
- Riego por evento: entre 10 y 15 litros en promedio: 12.5 L
- Frecuencia de riego: 2 veces por semana
- Duración del ciclo completo 135 días promedio de120 y 150

Consumo de agua por semana

$$Consumo\ semanal = Riego\ por\ evento \times Frecuencia\ semanal \quad \ldots(13)$$

$$Consumo\ semanal = 12.5L \times 2 = \frac{25L}{semana} por\ 1.44\ m^2 \quad \ldots(14)$$

Consumo total por ciclo completo

$$Semanas = \frac{7dias}{135dias} \approx 19.29\ semanas \quad \ldots(15)$$

Consumo por metro cuadrado

$$Consumo\ \ m^2 = \frac{482L}{1.44m^2} \approx \frac{334.7L}{m^2} \quad \ldots(16)$$

Este resultado es el consumo por metro cuadrado en el siclo de producción final del sistema.

En 15 días de consumo se calcularía de la siguiente manera:

$$Consumos\ en\ 15\ dias = \frac{Consumo\ total}{Dias\ total\ ciclo} \times 15 \quad \ldots(17)$$

$$Consumos\ en\ 15\ dias = \frac{\frac{334.7L}{m^2}}{135dias} \times 15dias \approx \frac{37.20L}{m^2} \quad \ldots(18)$$

4.5.6.2 Sistema IoT FVH

En cuanto al sistema IoT presentamos el área cubierta y el área de cada bandeja, también y así calculamos que cada bandeja tiene unas medidas de 0.3 por 0.4 m, se realiza el cálculo de cuanto sería el consumo de agua por área equivalente entre los dos sistemas, es decir el consumo de agua por ciclo del sistema en un metro cuadrado para realizar la comparativa de eficiencia del consumo de agua, estos datos para el cálculo equivalen a los datos registrados de la tabla 48.

Datos

- Número de bandejas: 96
- Dimensiones de cada bandeja: $0.3\ \text{m} \times 0.4\ \text{m} = 0.12m^2$
- Área total: $96 \times 0.12m^2 = 11.52m^2$
- Consumo total registrado en 6 semanas: 560 litros
- Ciclo completo de FVH: 15 días

Consumo por ciclo

Para esto realizamos el cálculo promedio acorde a los registros ya que en los registros tenemos tres ciclos de producción:

$$Consumo\ por\ ciclo = \frac{180 + 190 + 190}{3} = \frac{560}{3} \approx 186.67L \quad \ldots(19)$$

Cálculo de consumo por metro cuadrado

$$\text{Consumo por m}^2 = 186.67\text{L}/(11.52\text{m}^2) \approx 16.20\text{L/m}^2 \quad \textbf{...}(20)$$

4.5.6.3 Cálculo de eficiencia comparativa entre formas de producción

Ya con los datos obtenidos de las distintas pruebas y resultados registrados de ambos métodos se procede a calcular la eficiencia de un método respecto del otro en el periodo de 15 días de desarrollo de ambos sistemas.

Ahorro del consumo de agua

$$Ahorro\ (\%) = \frac{Consumo\ tradicional - Consumo\ FVH}{Consumo\ tradicional} \times 100 \quad \dots(21)$$

$$Ahorro\ (\%) = \frac{\frac{37.20L}{m^2} - \frac{16.20L}{m^2}}{\frac{37.20L}{m^2}} \times 100 \quad \dots(22)$$

$$Ahorro\ (\%) = \left(\frac{\frac{20.95L}{m^2}}{\frac{37.20L}{m^2}}\right) \times 100 = 56.4\% \quad \dots(23)$$

Ahora al saber que para el método tradicional se necesitan un aproximado de 135 días para poder ser cosechado se realiza el cálculo de eficiencia uno del otro ya que un ciclo de producción del método tradicional equivaldría a 9 ciclos de producción del sistema IoT de FVH.

$$Ahorro\ (\%) = \frac{Consumo\ tradicional - Consumo\ FVH}{Consumo\ tradicional} \times 100 \quad \dots(24)$$

$$Ahorro\ (\%) = \frac{\frac{334.7L}{m^2} - \frac{16.20L}{m^2}}{\frac{334.7L}{m^2}} \times 100 \quad \dots(25)$$

$$Ahorro\ (\%) = \left(\frac{\frac{318.5L}{m^2}}{\frac{334.7L}{m^2}}\right) \times 100 = 95.2\% \quad \dots(26)$$

4.6 Comparativa del Cultivo de FVH IoT con método tradicional

Tras los resultados obtenidos en el período de pruebas de ambas formas de producción, se evidencia en la **Tabla 49** la comparación de los datos entre sí, permitiendo comprender la utilidad y ventaja del sistema implementado frente a los métodos tradicionales de cultivo

Tabla 49

Tabla Comparativa de métodos de Cultivo

Parámetro de Evaluación	Sistema IoT FVH	Cultivo Tradicional	Comparación Técnica
Tiempo de Crecimiento	15 días	120 – 150 días	El FVH reduce el tiempo de producción, hasta 9 ciclos vs 1 ciclo tradicional.
Altura de Planta	18 – 20 cm	70 – 100 cm	El cultivo convencional alcanza mayor altura por desarrollo completo de la planta.
Producción de Biomasa	Alta densidad por bandeja	Dependiente del clima y calidad del suelo	FVH: uniformidad y producción controlada.
Uso de Agua por ciclo FVH	16.20 L/m²	37.20 L/m²	Ahorro del 56.4% en 15 días.
Uso de Agua por ciclo completo	16.20 L/m²	334.7 L/m² (1 ciclo tradicional)	Ahorro del 95.2**%** al comparar un ciclo tradicional vs uno FVH.
Dependencia Climática	Baja	Alta	El IoT elimina factores externos adversos. El tradicional está sujeto a lluvias,

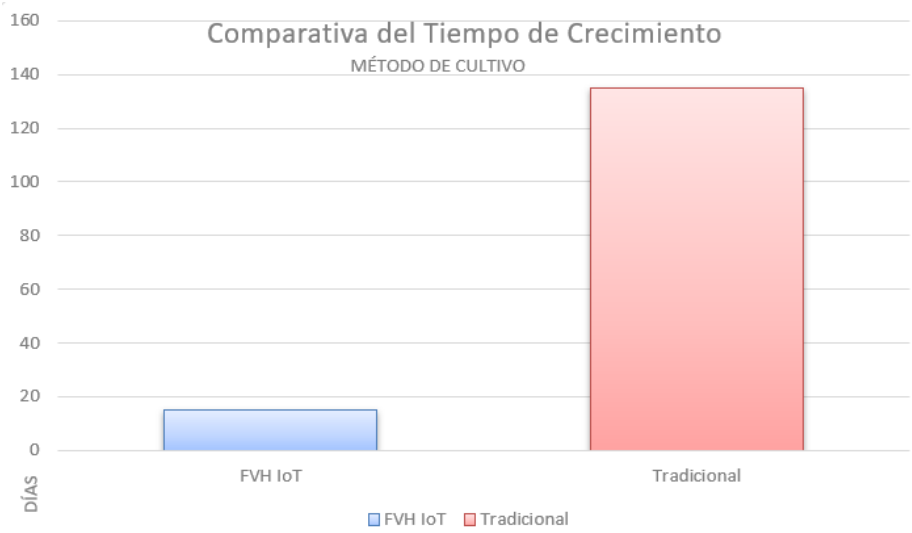
			temperatura, calidad del suelo.
Índice de Pérdida	Cercano a cero	Variable	Garantiza condiciones óptimas.
Desarrollo Radicular	Estera densa, blanca, aprovechable totalmente	Raíz profunda en suelo, no aprovechable	En FVH se aprovecha el 100% de la planta.
Calidad Nutricional	Alta.	Mayor contenido de materia seca y grano	FVH ideal para alimentación animal temprana y forraje fresco.
Frecuencia Producción	Hasta 24 cosechas/año (2 por mes)	1 0 2 cosechas al año	El FVH granatita la disponibilidad de alimento.
Uso de Suelo	No requiere suelo agrícola	Requiere terreno cultivable	En 1.44 m² de suelo tradicional, FVH produce efectivos 4 niveles en la misma área.
Área de producción	11.52 m²	1.44 m²	El FVH aprovecha 8 veces más el espacio.

Nota: *Diseño de autoría propia.*

El gráfico de tiempo de crecimiento mostrado en la **Figura 110** evidencia una diferencia significativa entre el sistema de Forraje Verde Hidropónico (FVH) de trigo desarrollado mediante IoT y el cultivo convencional de trigo. Mientras que el sistema hidropónico alcanzó su punto óptimo de cosecha en aproximadamente 15 días, el cultivo tradicional requiere alrededor de 135 días para completar su ciclo productivo. Esta reducción del tiempo de crecimiento representa una ventaja importante en términos de disponibilidad continua de alimento, optimización de recursos y aumento de la productividad anual, permitiendo realizar múltiples ciclos de producción en un mismo período de tiempo.

Figura 110

Tiempo de crecimiento de los cultivos



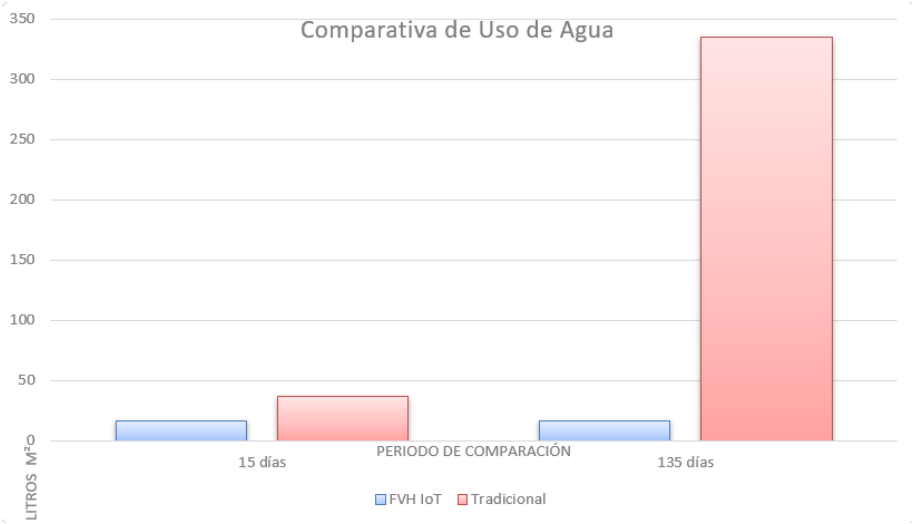
Nota: *Diseño de autoría propia.*

El gráfico de uso de agua mostrado en la **Figura 111** demuestra la eficiencia hídrica del sistema FVH automatizado en comparación con el cultivo convencional de trigo. El sistema hidropónico presentó un consumo aproximado de 16.20 litros por metro cuadrado a

comparación del tradicional que presenta 37.2L por metro cuadrado de igual manera se denota el consumo total por siclo de cada sistema.

Figura 111

Comparativa eficiencia del agua

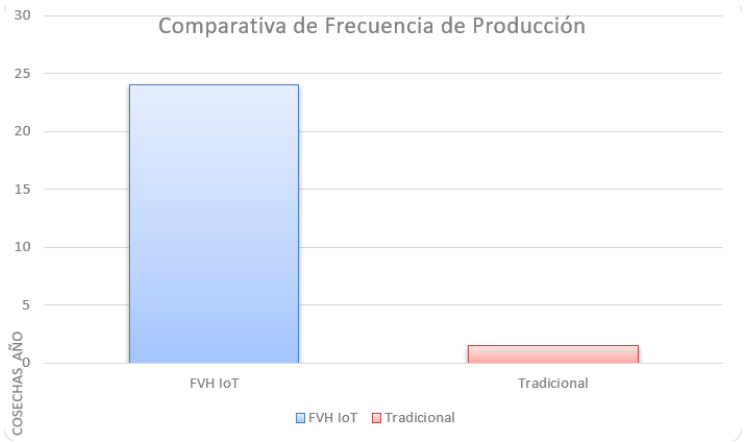


Nota: *Diseño de autoría propia.*

El gráfico de frecuencia de producción mostrado en la **Figura 112** indica que el sistema FVH de trigo permite alcanzar hasta 24 ciclos de producción total de las 96 bandejas durante el año, en comparación con el cultivo convencional, que generalmente produce entre 1 y 2 cosechas por año dependiendo de las condiciones climáticas y del suelo. Como resultado, el sistema hidropónico ofrece una producción más constante y eficiente, garantizando una mayor disponibilidad de forraje fresco para la alimentación animal durante todo el año.

Figura 112

Comparativa de frecuencia de cosechas

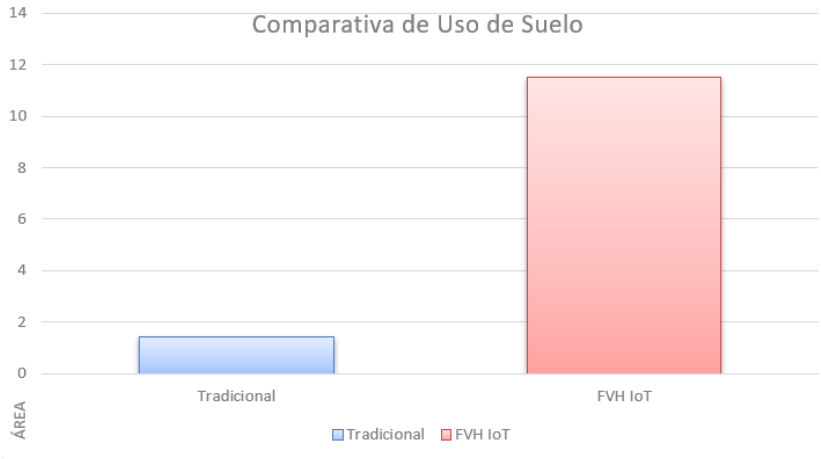


Nota: *Diseño de autoría propia.*

En la **Figura 113** se demuestra la eficiencia significativa del suelo ya que, dentro de una misma área de producción dentro del método tradicional, en el sistema IoT de FVH esta área de producción se multiplica por cuatro niveles aprovechando así de mejor manera los pocos espacios de producción. Y así en el área de 1.44 m cuadrados con el sistema Iot podemos tener hasta 12 m cuadrados de área productiva.

Figura 113

Uso del Suelo



Nota: *Diseño de autoría propia.*

4.7 Mantenimiento

El mantenimiento del sistema IoT implementado constituye un componente fundamental para garantizar la continuidad operativa, la precisión de las mediciones y la estabilidad del proceso automatizado. Debido a que el sistema integra dispositivos electrónicos, sensores ambientales, actuador hidráulico y plataformas de comunicación remota, resulta indispensable establecer procedimientos periódicos de mantenimiento preventivo y correctivo que permitan minimizar fallas y prolongar la vida útil del proyecto.

Las actividades de mantenimiento contemplan la recalibración de sensores, la actualización de software y plataformas de comunicación, la revisión del sistema hidráulico y eléctrico, así como la sanitización de los elementos físicos involucrados en la producción del FVH. Estas acciones permiten conservar la confiabilidad de los datos recolectados por el sistema, optimizar el funcionamiento del algoritmo de “Árbol de Decisión” y asegurar condiciones

adecuadas para el desarrollo del cultivo hidropónico, garantizando la disponibilidad del monitoreo remoto mediante tecnologías VPN y servicios de acceso seguro. La aplicación de este plan de mantenimiento permite que el sistema pueda adaptarse a futuras actualizaciones tecnológicas, cambios en las condiciones ambientales y nuevas necesidades productivas, contribuyendo a la sostenibilidad del proyecto a largo plazo.

4.7.1 Recalibración de Sensores

Los sensores implementados dentro de sistemas IoT agrícolas están expuestos continuamente a factores ambientales como humedad, variaciones térmicas, oxidación, acumulación de residuos minerales y desgaste electrónico, condiciones que pueden afectar progresivamente la precisión de las mediciones. Debido a ello, se estableció un procedimiento periódico de inspección y recalibración de los sensores utilizados en el sistema de monitoreo del cultivo hidropónico.

Dentro de las actividades preventivas se contempla la revisión visual del estado físico de los sensores de humedad, temperatura, nivel de agua y pH, verificando la ausencia de sulfatación, corrosión, fisuras, deformaciones o daños en los terminales de conexión. Asimismo, se realiza la limpieza superficial de los componentes electrónicos con el propósito de evitar acumulación de residuos que puedan alterar las lecturas analógicas o digitales del sistema.

El proceso de recalibración se ejecuta mediante códigos guardados por parte del programador, permitiendo ajustar nuevamente los valores de referencia utilizados por el algoritmo de control. Estas recalibraciones pueden realizarse después de determinados periodos de funcionamiento o cuando se detecten anomalías en las lecturas, tales como valores inestables, datos fuera del rango esperado o diferencias significativas respecto a mediciones manuales de control.

4.7.2 Actualización de métodos de comunicación

El sistema de monitoreo y control implementado depende de múltiples plataformas de software y servicios de comunicación remota, entre ellos el sistema operativo de la Raspberry Pi, Node-RED, servicios MQTT y la red privada virtual (VPN) utilizada para el acceso remoto. Debido a la naturaleza dinámica de estas tecnologías, los desarrolladores liberan constantemente actualizaciones para mejorar la estabilidad, compatibilidad y rendimiento de los servicios.

Por esta razón, se ideo un plan periódico de actualización del sistema operativo y de las plataformas utilizadas, con el objetivo de garantizar la continuidad funcional del sistema IoT y prevenir vulnerabilidades informáticas que puedan comprometer la comunicación entre dispositivos. Estas actualizaciones incluyen mejoras de seguridad, correcciones de errores, optimización del procesamiento de datos y compatibilidad con nuevas versiones de librerías y protocolos de comunicación.

De igual manera, se verificó periódicamente el correcto funcionamiento de los canales de comunicación entre sensores, servidores y aplicaciones móviles, asegurando que los flujos de datos enviados mediante Node-RED y MQTT se mantengan estables y sin interrupciones. En el caso del acceso remoto mediante VPN, se determinó la necesidad de realizar procesos de autenticación y renovación periódica de credenciales aproximadamente cada tres meses, garantizando que todos los dispositivos continúen integrados dentro de la red privada segura del sistema.

Estas actividades permiten conservar la disponibilidad del monitoreo remoto en tiempo real, facilitar el acceso a las alertas generadas por Telegram y asegurar la integridad de los datos

recopilados durante el proceso productivo del cultivo hidropónico.

4.7.3 Mantenimiento de materiales físicos de producción de FVH

El área física destinada a la producción de Forraje Verde Hidropónico requiere procedimientos permanentes de limpieza y sanitización con el propósito de evitar la proliferación de hongos, bacterias, algas y otros microorganismos que puedan afectar la calidad del cultivo o generar pérdidas en la producción.

Después de cada ciclo de cosecha, las bandejas hidropónicas, recipientes de almacenamiento, sistemas de riego y herramientas utilizadas durante el proceso productivo son sometidos a procesos de limpieza y desinfección conforme a las recomendaciones establecidas por el profesional agrónomo encargado de la supervisión técnica del proyecto. Estas actividades incluyen el retiro de residuos orgánicos, lavado con soluciones desinfectantes y secado adecuado de los componentes antes de iniciar un nuevo ciclo de siembra.

Asimismo, se realiza la inspección de las tuberías, conexiones hidráulicas, bomba de agua y estructuras de soporte, verificando la ausencia de fugas, obstrucciones o deterioro mecánico que puedan afectar el funcionamiento del sistema IoT de FVH. Este mantenimiento preventivo permite conservar condiciones higiénicas adecuadas dentro del entorno de producción y reducir significativamente el riesgo de contaminación del cultivo.

La correcta aplicación de estas medidas de mantenimiento físico contribuye a mantener la calidad nutricional del FVH de trigo, optimizar la producción de biomasa y asegurar la estabilidad operativa del sistema hidropónico supervisado mediante tecnologías IoT.

4.8 Costos del sistema

En esta sección se detallan los costos asociados a la implementación del sistema IoT desarrollado para el control automatizado del cultivo de forraje verde hidropónico (FVH) destinado a la crianza de cuyes en el cantón San Pedro de Huaca. Para el análisis económico se consideran los costos de hardware, infraestructura, ingeniería y software empleados durante las etapas de diseño, construcción, programación, implementación y pruebas del sistema.

4.8.1 Costo de Hardware

En esta sección se describen los componentes electrónicos y dispositivos empleados para la implementación del sistema IoT. Los costos presentados en la **Tabla 50** corresponden a valores referenciales obtenidos mediante adquisiciones realizadas en plataformas comerciales nacionales y distribuidores electrónicos a los cuales se visitó.

Tabla 50

Costos de Hardware

Equipos	**Cantidad**	**Precio Unitario**	**Subtotal**
Raspberry Pi 5	1	$ 170.00	$170.00
ESP32-WROOM-32D	1	$ 18.00	$ 18.00
Cargador Raspberry Pi 5	1	$ 15.00	$ 15.00
Cargadores ESP32	1	$ 10.00	$ 10.00
Sensor Capacitivo de Humedad V2.0	6	$ 2.75	$ 16.5
Sensor DHT22	1	$ 3.00	$ 3.00
Sensor BH1750	1	$ 5.00	$ 5.00
Sensor PH	1	$ 25.00	$ 25.00

Módulos Relé	1	$ 3.50	$ 3.50
Bomba de agua para riego	1	$ 150.00	$ 150.00
USB 128GB	1	$ 32.00	$ 32.00
Protoboard y conectores	1	$ 7.50	$ 7.5
Contactor	1	$ 25.00	$ 25.00
Elementos electrónicos varios	1	$ 40.00	$ 40.00
TOTAL			$ 520.5

Nota: *Diseño de autoría propia.*

4.8.2 Costo de Infraestructura

En la **Tabla 51** se describen los costos asociados a la infraestructura física utilizada para la implementación del sistema automatizado de producción de FVH. Se consideran materiales de construcción, sistema hidráulico, elementos de protección y estructura del invernadero.

Tabla 51

Costos de Infraestructura

Infraestructura	**Cantidad**	**Precio Unitario (USD)**	**Subtotal (USD)**
Manguera de 1½ pulgada	5 m	$1.20	$6.00
Conectores para manguera	2	$1.5	$3.00
Teflón	8	$1.00	$8.00
Abrazaderas metálicas	40	$ 0.50	$20.00
Cable UTP	25 m	$ 1.00	$25.00
Silicona liquida	1	$ 2.00	$2.00
Fungicida preventivo	1	$4.00	$4.00

Amarras plásticas	1	$4.00	$4.00
Cajas protectoras para nodos	1	$5.00	$5.00
Caja protectora Raspberry Pi 5	1	$7.00	$7.00
Manguera de riego por aspersión	16 m	$0.5	$8.00
Bandejas para FVH	100	$4.00	$400
Aspersores de nebulización	16	$2.50	$40.00
Tanque reservorio de agua	1	$20.00	$20.00
Estructura del invernadero	1	$200.00	$200.00
Base estructural	2	$200.00	$400.00
Plástico para recubrimiento	1	$300.00	$300.00
TOTAL			$**1452.00**

Nota: *Diseño de autoría propia.*

4.8.3 Costos de Ingeniería

Los costos de ingeniería consideran las actividades técnicas necesarias para el desarrollo del proyecto, incluyendo el diseño del sistema, programación, implementación, calibración, pruebas y documentación técnica del prototipo IoT todos estos presentados en la **Tabla 52.**

Tabla 52

Costos de Ingeniería

Actividades	**Cantidad**	**Precio Unitario**	**Subtotal**
Estudio de campo y análisis de requerimientos	1	$300.00	$300.00
Diseño e implementación del sistema IoT	1	$400.00	$400.00

Documentación y elaboración de tesis	1	$20.00	$20.00
Mantenimiento preventivo del sistema	1	$50.00	$50.00
TOTAL			$**770.00**

Nota: *Diseño de autoría propia.*

4.8.4 Costos de Software

En la **Tabla 53** se presentan las herramientas de software utilizadas durante el desarrollo del sistema. Debido a que todas las plataformas utilizadas corresponden a herramientas de código abierto (Open Source), no representan costos adicionales para el proyecto.

Tabla 53

Costos de Software

Software	Cantidad	Precio Unitario (USD)	Subtotal (USD)
Python	1	$0.00	$0.00
IDE Arduino	1	$0.00	$0.00
Node-RED	1	$0.00	$0.00
Visual Studio	1	$0.00	$0.00
Android Studio	1	$0.00	$0.00
Raspberry Pi OS	1	$0.00	$0.00
SQlite	1	$0.00	$0.00
Flutter	1	$0.00	$0.00
SketchUp	1	$0.00	$0.00
Tailscale	1	$0.00	$0.00
Draw.io	1	$0.00	$0.00

Telegram	1	$0.00	$0.00
TOTAL			$**0.00**

Nota: *Diseño de autoría propia.*

4.8.5 Costo Global del Sistema

Finalmente, en la Tabla 48 se presenta el costo total correspondiente a la implementación del sistema IoT para el control automatizado de producción de forraje verde hidropónico.

Tabla 54

Costos Globales

Descripción	**Subtotal (USD)**
Costos de Hardware	$ 520.5
Costos de Infraestructura	$1452.00
Costos de Ingeniería	$770.00
Costos de Software	0.00
TOTAL	**2742.00**

Nota: *Diseño de autoría propia.*

De acuerdo con la **Tabla 54**, el costo estimado para implementar el sistema desde cero es de aproximadamente 2742.00 dólares. Este valor representa una inversión inicial significativa para el productor. Sin embargo, es importante contrastar que, a largo plazo, el sistema permite aprovechar de manera óptima los espacios de producción, lo que contribuye a su viabilidad económica.

En el caso de productores de FVH que ya cuenten con instalaciones previamente construidas,

el costo de implementación del área tecnológica se reduce considerablemente a 1290.00 dólares, resultando una opción más viable y accesible para este tipo de productores que ya disponen de la infraestructura física necesaria.

4.9 Beneficios del sistema

La "IMPLEMENTACIÓN DE UN SISTEMA IOT PARA CONTROL EN EL DESARROLLO DE FORRAJE VERDE HIDROPÓNICO PARA LA CRIANZA DE CUYES EN EL CANTÓN SAN PEDRO DE HUACA" permitió evidenciar beneficios significativos tanto a nivel tecnológico como productivo, convirtiéndose en una alternativa viable para pequeños productores dedicados a la crianza de cuyes.

- El sistema logró reducir el tiempo requerido para la producción de forraje. Mientras el cultivo tradicional de trigo requiere entre 120 y 150 días para completar su ciclo productivo, el sistema FVH alcanzó condiciones óptimas de cosecha en aproximadamente 15 días, permitiendo aumentar la frecuencia de producción anual.
- La automatización del riego mediante sensores y algoritmos de inteligencia artificial permitió optimizar el consumo de agua, disminuyendo pérdidas ocasionadas por evaporación, infiltración y aplicaciones innecesarias de riego.
- El modelo basado en Árbol de Decisión incorporó criterios agronómicos relacionados con humedad, temperatura y luminosidad para determinar de forma autónoma para decidir cuando activar o detener el riego, mejorando la eficiencia de las decisiones de control.
- La producción en ambiente controlado permitió obtener un desarrollo uniforme de las plantas, favoreciendo una coloración homogénea y un sistema radicular saludable, aspectos asociados directamente con la calidad nutricional del forraje producido y

presentando un índice de mortalidad del producto casi nulo.

- El sistema IoT implementado permitió el monitoreo en tiempo real de variables críticas mediante plataformas de visualización, almacenamiento de datos y acceso remoto seguro, facilitando al productor mantener una supervisión permanente del cultivo de manera remota.
- La solución desarrollada constituye una alternativa tecnológica sostenible para pequeños productores, permitiendo producir grandes cantidades de alimento en espacios reducidos mediante estructuras verticales y sistemas hidropónicos automatizados.

CONCLUSIONES

- El análisis bibliográfico realizado y las consultas realizadas al ingeniero agrónomo de apoyo, permitió identificar las condiciones agronómicas necesarias para la producción eficiente de Forraje Verde Hidropónico y determinar que las tecnologías IoT e inteligencia artificial constituyen herramientas adecuadas para automatizar procesos agrícolas relacionados con el monitoreo y control ambiental.
- El sistema IoT propuesto fue diseñado e implementado con éxito, integrando un conjunto de sensores, un nodo sensor basado en el microcontrolador ESP32 y un Edge Gateway desplegado sobre una Raspberry Pi 5. La funcionalidad del sistema se alcanzó tras la superación de las pruebas técnicas necesarias, lo que permitió validar su operatividad en el contexto de la producción de Forraje Verde Hidropónico (FVH).
- El modelo de inteligencia artificial basado en Árbol de Decisión demostró ser una alternativa adecuada para la automatización del riego, identificando a la humedad de Las bandejas como la variable con mayor importancia en la toma de decisiones, complementada por información de temperatura, pH y luminosidad.
- El cultivo de trigo mediante la técnica de Forraje Verde Hidropónico (FVH) presentó un desarrollo uniforme, un adecuado crecimiento radicular y una coloración vegetal favorable, evidenciando condiciones apropiadas para la producción de alimento destinado a la crianza de cuyes en espacios reducidos, como apoyo para pequeños productores dedicados a esta actividad.
- La automatización del riego contribuyó a mejorar el aprovechamiento de los recursos hídricos, disminuyendo desperdicios y permitiendo mantener condiciones ambientales favorables para el desarrollo de FVH.

RECOMENDACIONES

- Debido a que el sistema está diseñado para crecer, se recomienda incorporar sensores con el fin de tener un control más preciso de los parámetros ambientales del sistema. Asimismo, se sugiere implementar actuadores adicionales para mantener un ambiente más controlado, con el fin de mejorar el control agronómico del FVH y optimizar las condiciones de crecimiento del forraje.
- Incrementar progresivamente el volumen de datos utilizados para entrenar el modelo de inteligencia artificial, los cuales son registrados durante el período de funcionamiento del sistema, con el propósito de mejorar la precisión de las reglas de decisión y adaptarlo a diferentes condiciones ambientales presentes y futuras que puedan suscitarse durante la operación del sistema IoT para producción de FVH.
- Garantizar una vida útil extensa del sistema es primordial, por lo que se recomienda realizar calibraciones periódicas de los sensores de humedad, temperatura, luminosidad y pH, asegurando así la confiabilidad de las mediciones empleadas por el sistema IoT para la toma de decisiones automatizada.
- Al ser un sistema IoT de gran escala, una de las recomendaciones fundamentales es incorporar sistemas de respaldo energético basados en paneles solares y bancos de baterías, con el fin de asegurar la continuidad operativa ante fallos de energía eléctrica. Cabe señalar que estos componentes no fueron implementados en el presente proyecto debido a su costo.
- Comenzar a desarrollar y entrenar el modelo para variar el grano que se producirá, dando oportunidad a otras variedades de semillas destinadas a FVH, con el objetivo de determinar su adaptabilidad y rendimiento en diferentes escenarios productivos.

BIBLIOGRAFÍA

Albuja, V., Andrade, J., Lucano, C., & Rodriguez, M. (2021). Comparativa de las ventajas de los sistemas hidropónicos como alternativas agrícolas en zonas urbanas. *Minerva*, *2*(4), 45–54. https://doi.org/10.47460/minerva.v2i4.26

Al-Emran, M., Malik, S. I., & Al-Kabi, M. N. (2020). A Survey of Internet of Things (IoT) in Education: Opportunities and Challenges. *Studies in Computational Intelligence*, *846*, 197–209. https://doi.org/10.1007/978-3-030-24513-9_12

Alvarez, C. A., & Noguera, G. D. F. (2023). Monitorización y análisis de variables ambientales en tiempo real. *E.T.S. de Ingeniería de Sistemas Informáticos*.

ARANO, C. (2007). *Hidroponía: algunas páginas de su historia*. https://www.horticom.com/revistasonline/horticultura/rhi58/24_33.pdf

Arumugam, S., Badrinath, R., Hernandez, A., & Holler, J. (2020). *Accelerating Industrial IoT Application Deployment through Reusable AI Components*. IEEE.

Astill, J., Dara, R. A., Fraser, E. D. G., Roberts, B., & Sharif, S. (2020). Smart poultry management: Smart sensors, big data, and the internet of things. In *Computers and Electronics in Agriculture* (Vol. 170). Elsevier B.V. https://doi.org/10.1016/j.compag.2020.105291

Balci, A., & Yildiz, O. (2020). *2020 7th International Conference on Energy Efficiency and Agricultural Engineering (EE & AE) : conference proceedings : 12th-14th November, 2020, Ruse, Bulgaria.*

Baye, W., Moges, F., & Andualem, D. (2024). Effect of hydroponic barley fodder on growth performance, carcass yield and carcass quality of cobb 500 broilers. *Heliyon*, *10*(13), e33909. https://doi.org/10.1016/J.HELIYON.2024.E33909

Bedolla, T. M., Espinosa, A., Palacios, O., Choix, F., Valle, F. de J. A., López Aguilar, D. R., Villavicencio, J. L. E., de La Pena, R. de L., Trujillo, A. G., Avila Serrano, N. Y., & Pérez, R. O. (2015). La irrigación con levaduras incrementa el contenido nutricional del forraje verde hidropónico de maíz. *Revista Argentina de Microbiologia*, *47*(3), 236–244. https://doi.org/10.1016/j.ram.2015.04.002

Belhakimi, A., Ghobrini, A., & Remram, Y. (2023). Development of Automatic Nutrient Dosing System for Hydroponics. *Proceedings - 2023 2nd International Conference on Electronics, Energy and Measurement, IC2EM 2023*. https://doi.org/10.1109/IC2EM59347.2023.10419584

Bonifaz, R. V. de los Á., Arellano, G. J. J., Malta, G. Y. I., & Torres, M. D. L. (2024). Hydroponic crops: a nutritional and sustainable alternative for the reproduction of guinea pigs (Cavia porcellus) in the coastal zone of Ecuador[Cultivos hidropônicos: uma alternativa nutricional e sustentável para a reprodução de porquinhos-da-índia (Cavia porcellus) na zona costeira do Equador][Cultivos hidropónicos: una alternativa nutricional y sostenible para la reproducción de cuyes (Cavia porcellus) en la zona costera de Ecuador]. *Sapienza*, *5*(2), e24042. https://doi.org/10.51798/sijis.v5i2.776

Bonilla, F. E. P. (2023). *MEJORAMIENTO DE LA EFICIENCIA DE UN SISTEMA DE RIEGO POR ASPERSIÓN BASADO EN REDES DE SENSORES Y REDES NEURONALES ARTIFICIALES PARA EL CULTIVO DE PASTO EN LA PARROQUIA DE TOCACHI PERTENECIENTE AL CANTÓN PEDRO MONCAYO*.

Bouziane, I., Belmokadem, H., & Moussaoui, M. (2023). A Review of Formal Security Verification of Common Internet of Things (IoT) Communication Protocols. *Colloquium in Information Science and Technology, CIST*, 334–341. https://doi.org/10.1109/CiSt56084.2023.10409999

Caldas, L., Castillo, D. A., Yanine, S., Prado, G., Lilian, M., Rosales, R., Luis, Q., & Vargas Leiva, D. (2019). *Diseño y construcción de sistemas acuapónicos a pequeña escala para familias de la región Piura Trabajo de Investigación*.

Camacho, J. G., Patiño, R., Luis Fernando Rodríguez, Diana Avilés, Mv., Jovanny Suquillo, María Nieto, Victoria López, César Azaquibay, Fausto Yumisaca, Pablo Garzón, J., César García, Antonio Guacapiña Revisores externos, Moreno Vargas, D., & de Revisión institucional Ing Jorge Rivadeneira Ing Diego Peñaherrera Ing Antonio Guacapiña Ing José Velásquez, C. (2022). *Diagnóstico del sistema de producción de cuyes en pequeños y medianos productores de la sierra del ecuador*. http://repositorio.iniap.gob.ec/handle/41000/5836

Castillo, A. C., Arellano Paúl Arellano, Patricio Peñafiel, Paola López, & Santiago Chiriboga. MSc. (2019). ACTUALIZACIÓN DEL PLAN DE DESARROLLO Y ORDENAMIENTO TERRITORIAL. *ACTUALIZACIÓN DEL PLAN DE DESARROLLO Y ORDENAMIENTO TERRITORIAL*. https://www.huaca.gob.ec/gadhuaca/images/pdf/PDYOT.pdf

Castillo, J., & Núñez, S. (2019). *Forraje verde hidropónico: una alternativa de producción ante el cambio climático Green forage hydroponic: a production alternative to climate change*. *5*(9). https://doi.org/10.5377/ribcc.v5i9.7947

Castro, P. (2002). SISTEMAS DE CRIANZA DE CUYES A NIVEL FAMILIAR-COMERCIAL EN EL SECTOR RURAL. *Benson Agriculture and Food Institute*.

Cevallos, M. E. H. (2021). *EVALUACIÓN DEL EFECTO DE BACTERIAS DIAZÓTROFAS EN EL RENDIMIENTO DE TRIGO (Triticum aestivum L.) CANTÓN ANTONIO ANTE, PROVINCIA DE IMBABURA”*.

Chandrasekaran, G., Kumar, N. S., Chokkalingam, A., Gowrishankar, V., Priyadarshi, N., &

Khan, B. (2023). IoT enabled smart solar water heater system using real time ThingSpeak IoT platform. *IET Renewable Power Generation*. https://doi.org/10.1049/rpg2.12760

Chanthakit, S., & Rattanapoka, C. (2018). Mqtt based air quality monitoring system using node MCU and node-red. *Proceeding of 2018 7th ICT International Student Project Conference, ICT-ISPC 2018*. https://doi.org/10.1109/ICT-ISPC.2018.8523891

Chen, D., Han, X., & Wang, W. (2010). Use of SQLite on embedded system. *Proceedings - 2010 International Conference on Intelligent Computing and Cognitive Informatics, ICICCI 2010*, 210–213. https://doi.org/10.1109/ICICCI.2010.79

Chima, V. M. E., & Luna, A. K. A. (2022, January 20). *Estudio de factibilidad para la producción y comercialización de cuy empacado al vacío en el Cantón El Tambo | Revista Científica FIPCAEC (Fomento de la investigación y publicación científico-técnica multidisciplinaria). ISSN : 2588-090X . Polo de Capacitación, Investigación y Publicación (POCAIP)*. https://www.fipcaec.com/index.php/fipcaec/article/view/519

Del Castillo, F. S., Del Carmen, M. P. E., Magaña, E. C., & Gómez, J. M. (2013). Producción de forraje hidropónico de trigo y cebada y su efecto en la ganancia de peso en borregos. *Revista Chapingo, Serie Horticultura*, *19*(4 SPEC.), 35–43. https://doi.org/10.5154/r.rchsh.2012.02.020

Dubey, P., & Tiwari, A. K. (2023). A Survey on Use of Amazon Web Services by Engineering Students. *Proceedings of IEEE 2023 5th International Conference on Advances in Electronics, Computers and Communications, ICAECC 2023*. https://doi.org/10.1109/ICAECC59324.2023.10560349

El Ghati, O., Alaoui-Fdili, O., Alioua, N., Chahbouni, O., & Bouarifi, W. (2023). An overview of the applications of AI-powered Visual IoT systems in agriculture.

Proceedings - 2023 IEEE International Conference on Advances in Data-Driven Analytics and Intelligent Systems, ADACIS 2023. https://doi.org/10.1109/ADACIS59737.2023.10424223

Elbasi, E., Mostafa, N., Alarnaout, Z., Zreikat, A. I., Cina, E., Varghese, G., Shdefat, A., Topcu, A. E., Abdelbaki, W., Mathew, S., & Zaki, C. (2023). Artificial Intelligence Technology in the Agricultural Sector: A Systematic Literature Review. In *IEEE Access* (Vol. 11, pp. 171–202). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/ACCESS.2022.3232485

Farinango, A., & Milton, J. (2023). *UNIVERSIDAD TÉCNICA DEL NORTE*.

Galindo, E. A., Perdomo, J. A., & Figueroa-García, J. C. (2020). Comparative study between multiclass support vector machines, neural networks and self-organized neuro-fuzzy inference system for classification problems. *Informacion Tecnologica*, *31*(1), 273–286. https://doi.org/10.4067/S0718-07642020000100273

García, Revelo Luis, Burbano, D. A., Chuquín Héctor Guillermo, Miranda Realpe, J., Luis Ramiro García Revelo, Diego Almeida Burbano, Héctor Guillermo Chuquín, & Jorge Miranda Realpe. (2018). *Estudio de potencialidades del cantón Huaca*. https://revistasdigitales.upec.edu.ec/index.php/visionempresarial/article/view/780

García, V. (2013). *Proyecto de factibilidad para la creación de una empresa productora y comerc ializadora de alimento de ganado menor y para mascotas (cuyes y conejos) he cho con forraje verde hidropónico fresco deshidratado en Tumbaco, Distrito Metropolitano de Quito*. http://dspace.ups.edu.ec/handle/123456789/5796

Gómez, M. (2012). *Evaluación del Forraje Verde Hidropónico de Maíz y Cebada, con Diferentes Dosis de Siembra para las Etapas de Crecimiento y Engorde de Cuyes*.

http://dspace.espoch.edu.ec/handle/123456789/1813

Google Earth. (2025). https://earth.google.com/web/@0.62950676,-77.71119101,2953.34618662a,251.89500741d,35y,24.71038666h,0t,0r/data=CgRCAggBMikKJwolCiExS1YzUXJKUnJKNk5vcGw3TENTaFRDd2xtcjczaDJvcnAgAToDCgEwQgIIAEoICPup47YHEAE

Guterres, A. (2023). *The-Sustainable-Development-Goals-Report-2023_Spanish*. https://www.un.org/sustainabledevelopment/es/infrastructure/

Hanen, A., Soumaya, B., & Faiez, G. (2015). *2015 IEEE ACS 12th International Conference of Computer Systems and Applications (AICCSA)*. IEEE.

Kabore, K. K. (n.d.). *Design and implementation of secure communication protocols for Internet of Things systems 1 st Didier BASSOLE*.

Kaskatiiski, N., & Boyanov, L. (2021). Efficiency of data exchange of IoT communication protocols. *International Conference Automatics and Informatics, ICAI 2021 Proceedings*, 358–361. https://doi.org/10.1109/ICAI52893.2021.9639627

Khujamatov, K., Khasanov, D., & Reypnazarov, E. (2019). *Research and Modelling Adaptive Management of Hybrid Power Supply Systems for Object Telecommunications based on IoT*.

Kostenko, S. I., Shamsutdinova, E. Z., Yu Novoselov, M., Piskovatsky, Y. M., & Tyurin, Y. S. (2021). Global climate change and the creation of adapted varieties of forage crops. *IOP Conference Series: Earth and Environmental Science*, *901*(1). https://doi.org/10.1088/1755-1315/901/1/012047

Ku, H. W., Tok, C. T., Suresh, A., & Ong, B. L. (2016). “Active” hydroponic greenhouse system to kick-start and augment reforestation program through carbon sequestration – an experimental and theoretical feasibility study. *Journal of Cleaner Production*, *129*,

637–646. https://doi.org/10.1016/J.JCLEPRO.2016.03.109

Laoyan, S. (2024). *Qué es la metodología waterfall y cuándo utilizarla • Asana*. https://asana.com/es/resources/waterfall-project-management-methodology

Larrea, H. I. (2022). *Efecto de dietas a base de forrajes arbustivos: chilca y eneldo en el rendimiento a la canal y características químicas de la carne de cuy*. https://repositorio.uta.edu.ec:8443/jspui/handle/123456789/34723

Llamas, I. (2022). *"IoT y agricultura 4.0: Control de cultivos en función de la conductividad eléctrica del suelo."*

Loza, G. G. A. (2021). *"FACTORES CRÍTICOS DE ÉXITO EN LA APLICACIÓN DE LA IOT AL SECTOR AGROPECUARIO."* https://dspace.ups.edu.ec/bitstream/123456789/20955/4/UPS-GT003393.pdf

Mercado libre. (2025a). *(20) Módulo de sensor de temperatura y humedad Arduino Dht22 Am2302 | Cuotas sin interés*. https://www.mercadolibre.com.ec/modulo-de-sensor-de-temperatura-y-humedad-arduino-dht22-am2302/p/MEC53274650#polycard_client=search-desktop&search_layout=grid&position=10&type=product&tracking_id=2bc05d00-4f43-4cee-834f-27cf8dc52e4a&wid=MEC745537588&sid=search

Mercado libre. (2025b). *(20) Modulo Esp32 D Devkit V4 Wifi Bluetooth 38 Pines Innovatech | Cuotas sin interés*. https://www.mercadolibre.com.ec/modulo-esp32-d-devkit-v4-wifi-bluetooth--38-pines-innovatech/up/MECU2438845698#polycard_client=search-desktop&search_layout=grid&position=8&type=product&tracking_id=d1609170-32aa-4e66-9933-ba052ad9adbe&wid=MEC528132710&sid=search

Mercado libre. (2025c). *(20) Raspberry Pi 5 (8 Gb) | Cuotas sin interés*. https://www.mercadolibre.com.ec/raspberry-pi-5-8-

gb/p/MEC32625223#polycard_client=search-desktop&search_layout=grid&position=7&type=product&tracking_id=55e7d6bb-d922-479f-b882-79a63d9ef9c3&wid=MEC572806608&sid=search

Mercado libre. (2025d). *Sensor Capacitivo Humedad Suelo V2 Arduino Pic | Cuotas sin interés*. https://www.mercadolibre.com.ec/sensor-capacitivo-humedad-suelo-v2-arduino-pic/up/MECU3482366102#polycard_client=search-desktop&search_layout=grid&position=3&type=product&tracking_id=926d144f-d9b3-4e59-a627-61f8478067ad&wid=MEC685532648&sid=search

Mercado libre. (2025e). *Sensor Ph + Acondicionador Arduino Raspberry | Cuotas sin interés*. https://www.mercadolibre.com.ec/sensor-ph-acondicionador-arduino-raspberry-electronics/p/MEC2048310208#polycard_client=search-desktop&search_layout=grid&position=4&type=product&tracking_id=88f4f89d-3a84-4cc0-b487-506f03a713ea&wid=MEC518421238&sid=search

Mercado libre. (2026). *Contactor 3 Polos 25a/40a 220v | MercadoLibre*. https://articulo.mercadolibre.com.ve/MLV-828110720-contactor-3-polos-25a40a-220v-_JM#polycard_client=search-desktop&search_layout=grid&position=4&type=item&tracking_id=f8e6a2cf-ba2f-40b7-99ad-cd9b42ddceb8&wid=MLV828110720&sid=search

Montes, A. T. (2012). *CRIANZA TECNIFICADA DE CUYES CONTENIDO*.

Novatronic. (2020). *Sensor de intensidad de luz GY302 BH1750 – Novatronic*. https://novatronicec.com/index.php/product/sensor-de-intensidad-de-luz-gy302-bh1750/

Núñez, T. O. P., & Guerrero, L. J. R. (2021). Forrajes hidropónicos: una alternativa para la alimentación de animales domésticos Hydroponic foods: an alternative for the feeding

of domestic animals Editado por: Selva Andina Research Society. In *Journal of the Selva Andina Animal Science ®. Bolivia. All rights reserved*.

Padilla, P. D. L. (2021). *ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO FACULTAD DE CIENCIAS PECUARIAS "FUENTES PROTEICAS PARA LA ALIMENTACIÓN DE CUYES EN GESTACIÓN Y LACTANCIA."*

Rahamathunnisa, U., Vivekanand, C. V., Chithras, T., Sreedevi, E., Kiran, V., & Rajendiran, M. (2024). Energy-Efficient Communication Protocols for IoT Devices. *Proceedings of 9th International Conference on Science, Technology, Engineering and Mathematics: The Role of Emerging Technologies in Digital Transformation, ICONSTEM 2024*. https://doi.org/10.1109/ICONSTEM60960.2024.10568674

Redondo, Q. D. (2023). *Desarrollo de un sistema de cultivo hidropónico vertical automatizado y remoto*. https://uvadoc.uva.es/handle/10324/60478

Reyes, D., Enríquez, E. I. A. , Uvidia Cabadiana, M. A., & Hernán IV. (2021). Análisis del manejo, producción y comercialización del cuy (Cavia porcellus L.) en Ecuador. *Dominio de Las Ciencias, ISSN-e 2477-8818, Vol. 7, Nº. 6, 2021 (Ejemplar Dedicado a: OCTUBRE 2021), Págs. 1004-1018*, *7*(6), 1004–1018. https://doi.org/10.23857/dc.v7i6.2377

Rouhiainen, L. (2018). *Inteligencia artificial : 101 cosas que debes saber hoy sobre nuestro futuro*. Alienta.

Salvi, S., Savitha, S., Sawargaonkar, S., Vishwas, B., Harshitha, T. K., & Pramod Jain, S. A. (2021). HydroIoT: An IoT and Edge Computing based Multi-Level Hydroponics System. *CSITSS 2021 - 2021 5th International Conference on Computational Systems and Information Technology for Sustainable Solutions, Proceedings*. https://doi.org/10.1109/CSITSS54238.2021.9683621

Saverio, R. (2016, November 8). *Looking for a Compass for Navigating the IoT Platform Landscape - IEEE Internet of Things*. https://iot.ieee.org/articles-publications/newsletter/november-2016/looking-for-a-compass-for-navigating-the-iot-platform-landscape.html?utm_source=chatgpt.com

Schoning, J., & Richter, M. L. (2021). AI-Based Crop Rotation for Sustainable Agriculture Worldwide. *2021 11th IEEE Global Humanitarian Technology Conference, GHTC 2021*, 142–146. https://doi.org/10.1109/GHTC53159.2021.9612460

Shinde, S., Patil, A., Sumedha, C., Sayali, D., & Ingleshwar, S. (2017). *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) : 10-11, February 2017*. IEEE.

Singh. (2009). *Database Systems : Concepts, Design and Applications*. 893.

Singh, V., & Singh, G. (2023). Development of AI Enabled Solution for Efficient Feature Enrichment from Multiple Data Sources: An Application in Precision Agriculture. *2023 International Conference on Artificial Intelligence and Applications, ICAIA 2023 and Alliance Technology Conference, ATCON-1 2023 - Proceeding*. https://doi.org/10.1109/ICAIA57370.2023.10169451

SQLite Home Page. (2025). https://sqlite.org/

Srdjan, K. B. P., & Carrez, F. (2014). *Designing IoT Architecture(s) A European Perspective*.

Sreyes, K., Anushka Xavier, K., Davis, D., & Jayapandian, N. (2022). Internet of Things and Cloud Computing Involvement Microsoft Azure Platform. *International Conference on Edge Computing and Applications, ICECAA 2022 - Proceedings*, 603–609. https://doi.org/10.1109/ICECAA55415.2022.9936126

Stolz, T., Huertas, M. E., & Mendoza, A. (2020). Assessment of air quality monitoring networks using an ensemble clustering method in the three major metropolitan areas of

Mexico. *Atmospheric Pollution Research*, *11*(8), 1271–1280. https://doi.org/10.1016/J.APR.2020.05.005

Telegram Messenger. (n.d.). Retrieved February 19, 2026, from https://telegram.org/

Tripathi, P., Miraz, M. H., & Joshi, S. (2023). Comparative Analysis of MongoDB and InfluxDB for Time Series Data Management in IoT Environments: A Study on Performance, Scalability, and Concurrency. *Proceedings - 2023 International Conference on Computing, Networking, Telecommunications and Engineering Sciences Applications, CoNTESA 2023*, 39–42. https://doi.org/10.1109/CoNTESA61248.2023.10384962

Ujibashi, Y., Nakamura, M., Tabaru, T., Hashida, T., Kawaba, M., & Harada, L. (2015). *Design of a Shared Memory Mechanism for Efficient Paralell Processing in PostgreSQL*.

Ullo, S. L., & Sinha, G. R. (2020). Advances in smart environment monitoring systems using iot and sensors. In *Sensors (Switzerland)* (Vol. 20, Number 11). MDPI AG. https://doi.org/10.3390/s20113113

Varela, A., & Ron, S. (2022, October 4). *Geografía y clima*. https://bioweb.bio/faunaweb/amphibiaweb/GeografiaClima/

Velarde, C. E. (2013). *"PROYECTO ANALISIS DE VULNERABILIDADES A NIVEL MUNICIPAL" PERFIL TERRITORIAL CANTON SAN PEDRO DE HUACA.*

Vishal, J., & Ashay, S. (2016). *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*. IEEE.

Yanchaliquin, J. (2022). *"FORRAJES HIDROPÓNICOS EN LA ALIMENTACIÓN DE CUYES."*

Zhu, Z., Yogev, U., Keesman, K. J., Rachmilevitch, S., & Gross, A. (2023). Integrated

hydroponics systems with anaerobic supernatant and aquaculture effluent in desert regions: Nutrient recovery and benefit analysis. *Science of The Total Environment*, *904*, 166867. https://doi.org/10.1016/J.SCITOTENV.2023.166867

ANEXOS

ANEXO 1: Validación de instrumento de investigación.

Instrucciones para el llenado del instrumento

El presente instrumento tiene como finalidad validar, mediante juicio de expertos, el contenido del cuestionario diseñado para evaluar el sistema inteligente para la producción de Forraje Verde Hidropónico (FVH) mediante IoT en el cantón San Pedro de Huaca. Su colaboración permitirá asegurar la coherencia, pertinencia y claridad de los ítems planteados.

Por favor, revise con atención cada uno de los ítems del instrumento y seleccione la opción que mejor refleje su nivel de acuerdo, utilizando la siguiente escala:

Valor	Criterio
1	Totalmente en desacuerdo
2	En desacuerdo
3	Ni de acuerdo ni en desacuerdo
4	De acuerdo
5	Totalmente de acuerdo

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE TELECOMUNICACIONES

FORMATO DE VALIDACIÓN DE INSTRUMENTO DE INVESTIGACIÓN

(junio del 2026)

• Para garantizar la más amplia participación en las conversaciones entre los involucrados en el desarrollo del proyecto

"IMPLEMENTACIÓN DE UN SISTEMA IOT PARA CONTROL EN EL DESARROLLO DE FORRAJE VERDE HIDROPÓNICO PARA LA CRIANZA DE CUYES EN EL CANTÓN SAN PEDRO DE HUACA", el Ing. Jairo Geovanny Hernández Rosales y el equipo de investigación presentan las propuestas de requerimientos del sistema a la mesa de validación, con el objetivo de garantizar la pertinencia técnica y científica del diseño planteado.

• Las propuestas se estructuran en función de los ejes temáticos definidos para el proceso de validación técnica:

- Requerimientos operacionales y de usuario.
- Requerimientos del sistema.
- Requerimientos de arquitectura del sistema.

Estos ejes buscan contribuir a la consolidación de una solución tecnológica eficiente, replicable y sostenible, orientada al control y supervisión inteligente de la producción de Forraje Verde Hidropónico (FVH) para la crianza de cuyes.

TABLA DE VALIDACIÓN DE REQUERIMIENTOS DEL SISTEMA IoT-FVH

N°	Ítem de Validación	Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
1	El sistema se adapta a las condiciones controladas del invernadero para FVH sin intervención constante.	☐	☐	☐	☑	☐
2	La interfaz Node-RED permite visualizar claramente temperatura, humedad, luminosidad, humedad del sustrato y pH.	☐	☐	☐	☐	☑
3	El sistema notifica fallos o alertas del cultivo (sequía, saturación, pH) mediante Telegram de forma efectiva.	☐	☐	☐	☑	☐
4	La iluminación empleada estimula el crecimiento foliar y desarrollo del cultivo de trigo hidropónico.	☐	☐	☐	☐	☑
5	El algoritmo de Árbol de Decisión clasifica el estado del cultivo y determina el riego óptimo en tiempo real.	☐	☐	☐	☐	☑
6	Los sensores (DHT22, capacitivo v2.0, BH1750, PH-4502C) responden con precisión a cambios ambientales.	☐	☐	☑	☐	☐
7	El sistema permite comparar el rendimiento entre cultivo tradicional y cultivo hidropónico automatizado FVH IoT.	☐	☐	☐	☑	☐
8	El almacenamiento SQLite asegura la integridad y disponibilidad de la información histórica del cultivo.	☐	☐	☐	☐	☑
9	Los actuadores se activan correctamente según decisiones del modelo de IA.	☐	☐	☐	☑	☐
10	El sistema opera durante todo el ciclo del cultivo (12-15 días) sin interrupciones críticas.	☐	☐	☐	☑	☐
11	El diseño facilita el mantenimiento y ampliaciones a otros forrajes (avena, cebada) o cultivos.	☐	☐	☐	☐	☑
12	El usuario puede interpretar correctamente las predicciones y alertas de la aplicación móvil FVH Control App.	☐	☐	☐	☐	☑
13	El sistema optimiza el uso del espacio en el invernadero mediante	☐	☐	☐	☐	☑

	estantes verticales para maximizar producción.					
14	La fuente de alimentación es adecuada y segura para todos los componentes (ESP32, Raspberry Pi 5, sensores, bomba).	☐	☐	☑	☐	☐
15	Se cumplen los objetivos de mejora en producción y calidad nutricional del FVH frente al método tradicional.	☐	☐	☐	☐	☑

Ing. Luis Edilberto Suárez Zambrano

Tutor de Tesis

Ing. Jairo Geovanny Hernández Rosales

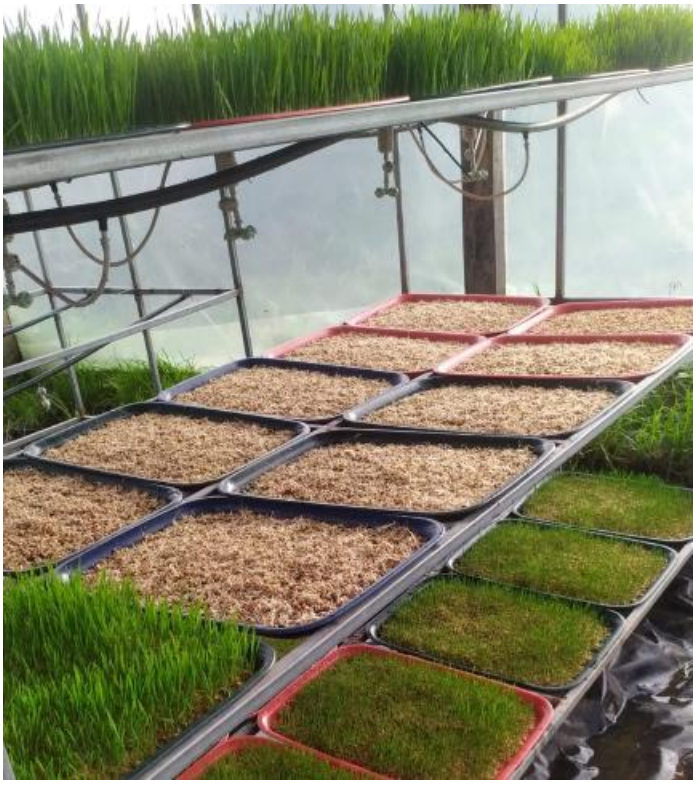
Asesor Agrónomo

Sr. Erik Leonardo Cuasquer Cuasapaz

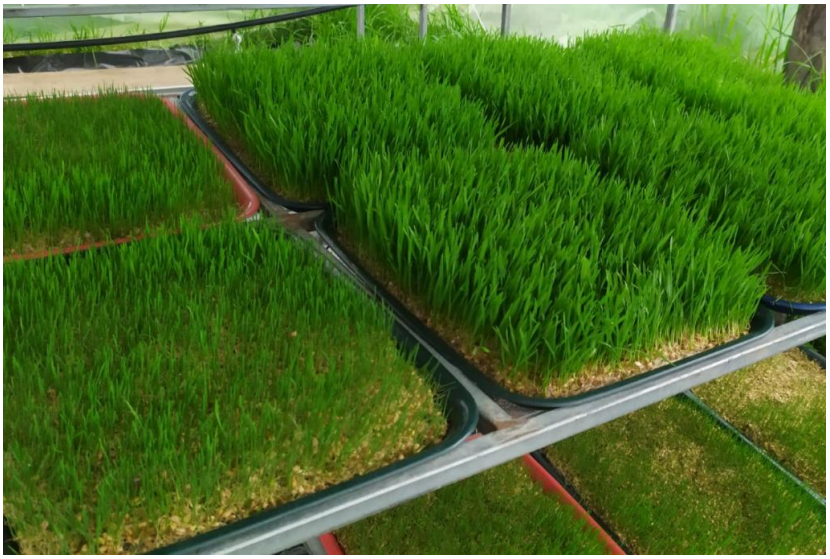
Tesista

ANEXO 2: Funcionamiento del sistema para controlar el desarrollo del fvh.

Siembra de FVH.



Desarrollo Fvh.



Cosecha FVH.



Alimentación de Cuyes con FVH.



ANEXO 3: Datasheets de las placas del sistema.

ESP32-WROOM-32D & ESP32-WROOM-32U

Datasheet Version 2.6

NOT RECOMMENDED FOR NEW DESIGNS (NRND)

ESPRESSIF

www.espressif.com

1 Module Overview

Note:

Check the link or the QR code to make sure that you use the latest version of this document:
https://espressif.com/documentation//esp32-wroom-32d_esp32-wroom-32u_datasheet_en.pdf

1.1 Features

CPU and On-Chip Memory

- ESP32-D0WD embedded, Xtensa dual-core 32-bit LX6 microprocessor, up to 240 MHz
- 448 KB ROM
- 520 KB SRAM
- 8 KB SRAM in RTC

Wi-Fi

- 802.11b/g/n
- Bit rate: 802.11n up to 150 Mbps
- A-MPDU and A-MSDU aggregation
- 0.4 μs guard interval support
- Center frequency range of operating channel: 2412 ~ 2484 MHz

Bluetooth®

- Bluetooth V4.2 BR/EDR and Bluetooth LE specification
- Class-1, class-2 and class-3 transmitter
- AFH
- CVSD and SBC

Peripherals

- Up to 32 GPIOs
  - 5 strapping GPIOs
- SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, TWAI® (compatible with ISO 11898-1, i.e. CAN Specification 2.0)

Integrated Components on Module

- 40 MHz crystal oscillator
- 4 MB SPI flash

Antenna Options

- ESP32-WROOM-32D: On-board PCB antenna
- ESP32-WROOM-32U: External antenna via a connector

Operating Conditions

- Operating voltage/Power supply: 3.0 ~ 3.6 V
- Operating ambient temperature: –40 ~ 85 °C

Certification

- RF certification: See certificates
- Green certification: REACH/RoHS

Test

- HTOL/HTSL/uHAST/TCT/ESD

1.2 Series Comparison

ESP32-WROOM-32D and ESP32-WROOM-32U are powerful, generic Wi-Fi + Bluetooth® + Bluetooth LE MCU modules that target a wide variety of applications, ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming and MP3 decoding.

2.2 Pin Description

The module has 38 pins. See pin definitions in Table 3 *Pin Description*.

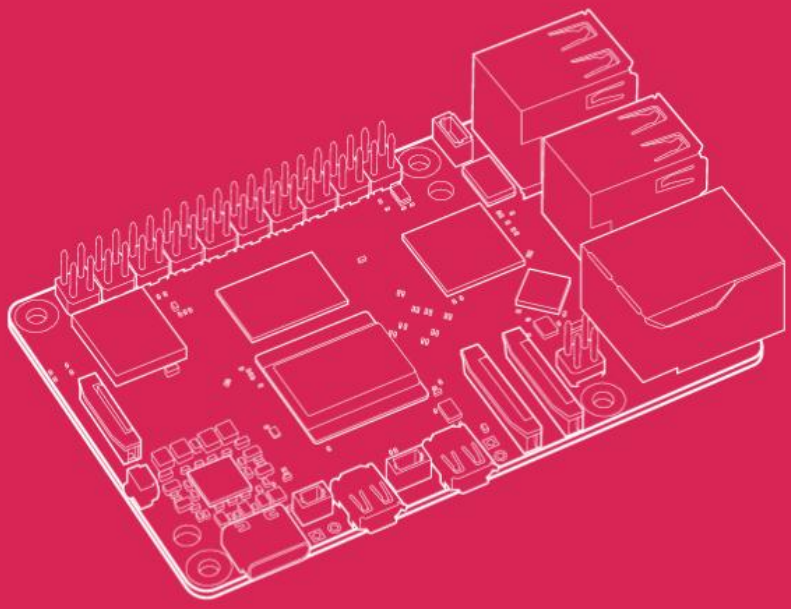
Table 3: Pin Definitions

Name	No.	Type	Function
GND	1	P	Ground
3V3	2	P	Power supply

Name	No.	Type	Function
EN	3	I	Module-enable signal. Active high.
SENSOR_VP	4	I	GPIO36, ADC1_CH0, RTC_GPIO0
SENSOR_VN	5	I	GPIO39, ADC1_CH3, RTC_GPIO3
IO34	6	I	GPIO34, ADC1_CH6, RTC_GPIO4
IO35	7	I	GPIO35, ADC1_CH7, RTC_GPIO5
IO32	8	I/O	GPIO32, XTAL_32K_P (32.768 kHz crystal oscillator input), ADC1_CH4, TOUCH9, RTC_GPIO9
IO33	9	I/O	GPIO33, XTAL_32K_N (32.768 kHz crystal oscillator output), ADC1_CH5, TOUCH8, RTC_GPIO8
IO25	10	I/O	GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0
IO26	11	I/O	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
IO27	12	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
IO14	13	I/O	GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2
IO12	14	I/O	GPIO12, ADC2_CH5, TOUCH5, RTC_GPIO15, MTDI, HSPIQ, HS2_DATA2, SD_DATA2, EMAC_TXD3
GND	15	P	Ground
IO13	16	I/O	GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER
SHD/SD2*	17	I/O	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
SWP/SD3*	18	I/O	GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD
SCS/CMD*	19	I/O	GPIO11, SD_CMD, SPICS0, HS1_CMD, U1RTS
SCK/CLK*	20	I/O	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS
SDO/SD0*	21	I/O	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS
SDI/SD1*	22	I/O	GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
IO15	23	I/O	GPIO15, ADC2_CH3, TOUCH3, MTDO, HSPICS0, RTC_GPIO13, HS2_CMD, SD_CMD, EMAC_RXD3
IO2	24	I/O	GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPIWP, HS2_DATA0, SD_DATA0
IO0	25	I/O	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
IO4	26	I/O	GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
IO16	27	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
IO17	28	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
IO5	29	I/O	GPIO5, VSPICS0, HS1_DATA6, EMAC_RX_CLK
IO18	30	I/O	GPIO18, VSPICLK, HS1_DATA7
IO19	31	I/O	GPIO19, VSPIQ, U0CTS, EMAC_TXD0
NC	32	-	-
IO21	33	I/O	GPIO21, VSPIHD, EMAC_TX_EN

Raspberry Pi 5

Published April 2026



HDMI™
HIGH-DEFINITION MULTIMEDIA INTERFACE
The terms HDMI, HDMI High-Definition

Specification

Processor: Broadcom BCM2712 2.4GHz quad-core 64-bit Arm Cortex-A76 CPU, with Cryptographic Extension, 512KB per-core L2 caches, and a 2MB shared L3 cache

Features:

- VideoCore VII GPU, supporting OpenGL ES 3.1, Vulkan 1.2
- Dual 4Kp60 HDMI® display output with HDR support
- 4Kp60 HEVC decoder
- LPDDR4X-4267 SDRAM (options for 1GB, 2GB, 4GB, 8GB and 16GB)
- Dual-band 802.11ac Wi-Fi®
- Bluetooth 5.0 / Bluetooth Low Energy (BLE)
- microSD card slot, with support for high-speed SDR104 mode
- 2 × USB 3.0 ports, supporting simultaneous 5Gbps operation
- 2 × USB 2.0 ports
- Gigabit Ethernet, with PoE+ support (requires separate PoE+ HAT)
- 2 × 4-lane MIPI camera/display transceivers
- PCIe 2.0 x1 interface for fast peripherals (requires separate M.2 HAT or other adapter)
- 5V/5A DC power via USB-C, with Power Delivery support
- Raspberry Pi standard 40-pin header
- Real-time clock (RTC), powered from external battery
- Power button

MTBF[1] Ground Benign: 93 800 hours

Operating temperature: 0°C to 70°C

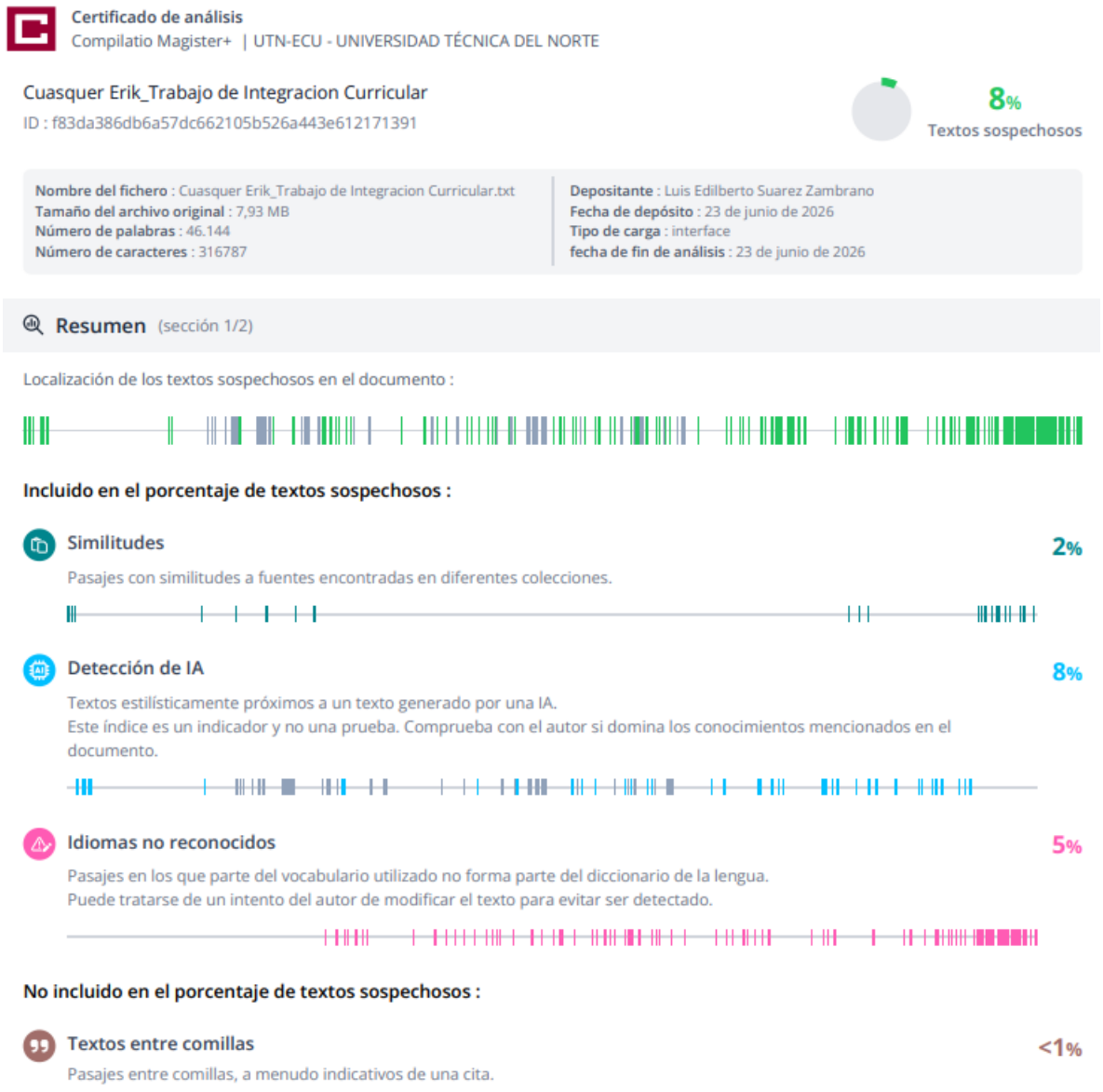
Production lifetime: Raspberry Pi 5 will remain in production until at least January 2036

Compliance: For a full list of local and regional product approvals, please visit pip.raspberrypi.com

List price:

1GB	**$45**
2GB	**$65**
4GB	**$110**
8GB	**$175**
16GB	**$305**

Anexo 5: Informe compilatio



Certificado de análisis
Compilatio Magister+ | UTN-ECU - UNIVERSIDAD TÉCNICA DEL NORTE

Cuasquer Erik_Trabajo de Integracion Curricular
ID : f83da386db6a57dc662105b526a443e612171391

8%
Textos sospechosos

Nombre del fichero : Cuasquer Erik_Trabajo de Integracion Curricular.txt
Tamaño del archivo original : 7,93 MB
Número de palabras : 46.144
Número de caracteres : 316787

Depositante : Luis Edilberto Suarez Zambrano
Fecha de depósito : 23 de junio de 2026
Tipo de carga : interface
fecha de fin de análisis : 23 de junio de 2026

Resumen (sección 1/2)

Localización de los textos sospechosos en el documento :

Incluido en el porcentaje de textos sospechosos :

Similitudes 2%
Pasajes con similitudes a fuentes encontradas en diferentes colecciones.

Detección de IA 8%
Textos estilísticamente próximos a un texto generado por una IA.
Este índice es un indicador y no una prueba. Comprueba con el autor si domina los conocimientos mencionados en el documento.

Idiomas no reconocidos 5%
Pasajes en los que parte del vocabulario utilizado no forma parte del diccionario de la lengua.
Puede tratarse de un intento del autor de modificar el texto para evitar ser detectado.

No incluido en el porcentaje de textos sospechosos :

Textos entre comillas <1%
Pasajes entre comillas, a menudo indicativos de una cita.