

**UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO**  
**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**  
**ESCUELA PROFESIONAL DE INGENIERÍA EN**  
**COMPUTACIÓN E INFORMÁTICA**



**TESIS**

**Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno  
para Controlar Nivel de Agua en la Quinta Fernández**

**Presentado para optar el Título Profesional de:  
Ingeniero (a) en Computación e Informática**

**Investigadores:**

**Bach. Fernández Rodríguez Fanny Eliza  
Bach Ramos Gamarra Jair Edú**

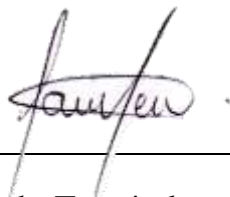
**Asesor:**

**Dra. Ing. Maquen Niño Gisela Luisa Elena**

**Lambayeque, 2022**

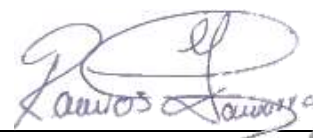
“Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno  
para Controlar Nivel de Agua en la Quinta Fernández”

Presentado por:



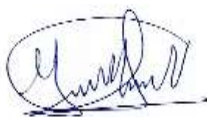
---

Bach. Fernández  
Rodríguez Fanny Eliza



---

Bach. Ramos Gamarra  
Jair Edú



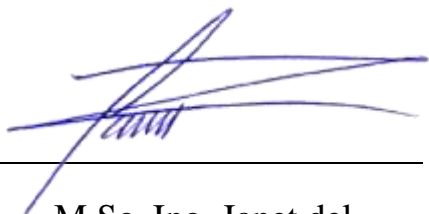
---

Dra. Ing. Gisella Luisa  
Elena Maquén Niño

**Asesora**

“Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno  
para Controlar Nivel de Agua en la Quinta Fernández”

Aprobado por:



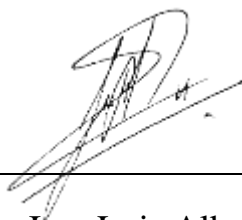
M.Sc. Ing. Janet del  
Rosario Aquino Lalupú

**Presidenta**



Mg. Ing. Percy Javier  
Celis Bravo

**Secretario**



M.Sc. Ing. Luis Alberto  
Reyes Lescano

**Vocal**

### CONSTANCIA DE VERIFICACIÓN DE ORIGINALIDAD

Yo, Gisella Luisa Elena Maquen Niño asesora del bachiller Fernández Rodríguez Fanny Eliza y Ramos Gamarra Jair Edú.

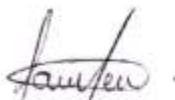
Titulada: Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para Controlar Nivel de Agua en la Quinta Fernández, luego de la revisión exhaustiva del documento constato que la misma tiene un índice de similitud de 17 %, verificable en el reporte de similitud del programa Turnitin.

El suscrito analizó dicho reporte y concluyó que cada una de las coincidencias detectadas no constituyen plagio. A mi leal saber y entender la tesis cumple con todas las normas para el uso de citas y referencias establecidas por la Universidad Nacional Pedro Ruiz Gallo.

Lambayeque, 06 de septiembre del 2023



GISELLA LUISA ELENA  
MAQUEN NIÑO  
DNI: 41747228  
ASESOR



FANNY ELIZA  
FERNÁNDEZ RODRÍGUEZ  
DNI: 75719188  
TESISTA



JAIR EDÚ RAMOS  
GAMARRA  
DNI: 73145554  
TESISTA



**UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO**  
**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**  
**DECANATO**  
 Ciudad Universitaria - Lambayeque



**ACTA DE SUSTENTACIÓN VIRTUAL N° 026-2023-D/FACFyM**

Siendo las 11:00 am del día 15 de junio del 2023, se reunieron vía plataforma virtual, <https://meet.google.com/rik-modd-vpf?hs=224> los miembros del jurado evaluador de la Tesis titulada: **PROTOTIPO DE SISTEMA EN TIEMPO REAL DE BEBEDEROS DE GANADO VACUNO PARA CONTROLAR NIVEL DE AGUA EN LA QUINTA FERNÁNDEZ**

Designados por Resolución N° 883-2021-VIRTUAL-D/FACFyM de fecha 30 de noviembre de 2021  
 Con la finalidad de evaluar y calificar la sustentación de la tesis antes mencionada, conformada por los siguientes docentes:

<b>M.Sc. Ing. Janet del Rosario Aquino Lalupú</b>	Presidente
<b>Mg. Ing. Percy Javier Celis Bravo</b>	Secretario
<b>M.Sc. Ing. Luis Alberto Reyes Lescano</b>	Vocal

La tesis fue asesorada por (el) (la) Dra. Ing. Gisella Luisa Elena Maquén Niño nombrado por Resolución N° 883-2021-VIRTUAL-D/FACFyM de fecha 30 de noviembre de 2021

El Acto de Sustentación fue autorizado por Resolución N° 440-2023-VIRTUAL -D/FACFyM de fecha 1 de junio de 2023

La Tesis fue presentada y sustentada por los Bachilleres: Fernández Rodríguez Fanny Eliza y Ramos Gamarra Jair Edú y tuvo una duración de 50 minutos.

Después de la sustentación, y absueltas las preguntas y observaciones de los miembros del jurado se procedió a la calificación respectiva, otorgándole el Calificativo de 17 Diecisiete en la escala vigesimal, mención Bueno.

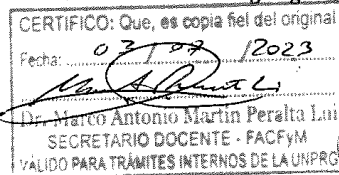
Por lo que queda (n) apto (s) para obtener el Título Profesional de **Ingeniero (a) en Computación e Informática** de acuerdo con la Ley Universitaria 30220 y la normatividad vigente de la Facultad de Ciencias Físicas y Matemáticas y la Universidad Nacional Pedro Ruiz Gallo.

Siendo las 12:21 pm se dio por concluido el presente acto académico, dándose conformidad al presente acto con la firma de los miembros del jurado.

**M.Sc. Ing. Janet del Rosario Aquino Lalupú**  
 Presidente

**Mg. Ing. Percy Javier Celis Bravo**  
 Secretario

**M.Sc. Ing. Luis Alberto Reyes Lescano**  
 Vocal



**Dra. Ing. Gisella Luisa Elena Maquén Niño**  
 Asesor

### **Declaración jurada de Originalidad**

Yo, Fanny Eliza Fernández Rodríguez investigadora principal, y Gisella Luisa Elena Maquén Niño asesora del trabajo de investigación “Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para Controlar Nivel de Agua en la Quinta Fernández”, declaramos bajo juramento que este trabajo no ha sido plagiado, ni contiene datos falsos. En caso se demostrará lo contrario, asumo responsablemente la anulación de este informe y por ende el proceso administrativo a que hubiera lugar. Que puede conducir a la anulación del título o grado emitido como consecuencia de este informe.

Lambayeque, 15 de junio del 2023

Nombre Investigador (es)

Fanny Eliza Fernández Rodríguez

Jair Edú Ramos Gamarra

Nombre Asesora

Gisella Luisa Elena Maquén Niño

## Dedicatoria

*A mis padres, Apolinar Fernández Guevara y Fanny Rodríguez Cabrera, por su amor incondicional, por apoyarme y motivarme a seguir, por darme la oportunidad de tener una carrera universitaria, sin ellos no lo hubiera logrado.*

*A mis abuelos, hermanos y tíos por su apoyo para seguir adelante día a día.*

***Fanny Eliza Fernández Rodríguez***

*A mi padre Felipe Ramos Montenegro y a mi madre Luz Gamarra Estrada, quienes me apoyaron toda mi etapa universitaria.*

*A mis hermanos Freddy y Blanca porque siempre me han brindado sus consejos profesionales a lo largo de mi carrera profesional.*

***Jair Edú Ramos Gamarra***

## **Agradecimiento**

En primer lugar, agradecer a Dios por permitirnos terminar nuestra etapa universitaria y convertirnos en profesionales.

A nuestra familia, por su apoyo constante y respaldo frente a las adversidades que se nos presentaron.

A nuestra asesora Dra. Ing. Gisella Luisa Elena Maquén Niño por sus enseñanzas, por guiarnos y motivarnos a lo largo del desarrollo de nuestro proyecto, y sobre todo por su paciencia infinita.

Finalmente agradecemos a quien lee este apartado y más de nuestra tesis, por permitir que nuestra investigación forme parte de su conocimiento.



## Resumen

El presente proyecto ha sido desarrollado con la finalidad de implementar un prototipo de Sistema en tiempo real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández.

El tipo de investigación es Investigación Aplicada Tecnológica, de acuerdo con su nivel de profundización será una Investigación Explicativa, dependiendo el tipo de datos utilizados, se aplicará la Investigación Cuantitativa, dependiendo del grado de manipulación de variables, es una investigación No Experimental, con Diseño Transeccional y según el tipo de inferencia hemos considerado que es una Investigación Hipotética-deductiva.

Para lograr el objetivo propuesto se ha diseñado un prototipo, el cual está compuesto por diversas conexiones eléctricas y sensores, teniendo como pieza principal la placa ESP32, la cual tiene tecnología WI-FI integrada que nos permite conectarla con la aplicación Blynk y tener el control de diversas funcionalidades del bebedero en tiempo real.

Después de realizar las pruebas respectivas al prototipo de bebedero para ganado vacuno, se configuró como límite inferior del nivel del agua el veinte por ciento de la capacidad del bebedero y como límite superior el cien por ciento de su capacidad, así mismo se configuro como grado de turbidez aceptable la salida del sensor de turbidez menor a tres. Se obtuvo como resultado que el bebedero funcionó correctamente todas las veces que llegaba a los límites establecidos del nivel del agua, de tal manera que cuando alcanzaba el límite superior automáticamente cerraba la válvula solenoide por orden del relé, cortando el flujo de agua. Cuando se encontraba en el límite inferior, el sistema automáticamente abría la válvula solenoide por orden del relé, empezando el llenado del bebedero. Cuando el sistema detectaba que el agua esta turbia (valor de salida del sensor de turbidez, mayor o igual a tres), enviaba un correo al ganadero donde indica que se ha sobrepasado los niveles de turbidez y que necesita ser limpiado, el ganadero tiene dos opciones: activar desde la plataforma Blynk de su celular la opción de vaciado del bebedero, o la otra opción es que el ganadero presione un botón que se encuentra en el prototipo, e inicie el proceso de vaciado. Normalmente el ganadero, cuando detecta que esta turbia, deja un porcentaje del agua para lavar manualmente el bebedero.

**Palabras clave:** Bebedero para Ganado Vacuno, Bebedero Automático, Sistema en Tiempo Real, Control de Nivel de Agua, Sensor de Turbidez.

### **Abstract**

This project has been developed with the purpose of implementing a prototype of a real time system of cattle watering troughs to control water level in the fifth Fernandez.

The type of research is Applied Technological Research, according to its level of depth it will be an Explanatory Research, depending on the type of data used, Quantitative Research will be applied, depending on the degree of manipulation of variables, it is a Non Experimental research, with Transectional Design and according to the type of inference we have considered it to be a Hypothetical-deductive Research.

To achieve the proposed objective, a prototype has been designed, which is composed of various electrical connections and sensors, having as main piece the ESP32 board, which has integrated WI-FI technology that allows us to connect it with the Blynk application and have control of various functionalities of the drinking fountain in real time.

After performing the respective tests to the prototype of the drinking trough for cattle, the lower limit of the water level was set as twenty percent of the capacity of the drinking trough and the upper limit as one hundred percent of its capacity, as well as the acceptable turbidity level was set as the output of the turbidity sensor of less than two. As a result, the trough functioned correctly every time it reached the established water level limits, in such a way that when it reached the upper limit it automatically closed the solenoid valve by order of the relay, cutting off the water flow. When it reached the lower limit, the system automatically opened the solenoid valve on relay command, starting the filling of the sprue. When the system detected that the water was cloudy (turbidity sensor output value greater than or equal to two), it sent an email to the farmer indicating that the turbidity levels had been exceeded and that it needed to be cleaned, the farmer had two options: activate from the Blynk platform of his cell phone the option of emptying the trough, or the other option is that the farmer presses a button on the prototype, and initiates the emptying process. Normally, when the farmer detects that the water is cloudy, he leaves a percentage of the water to manually wash the drinking trough.

**Key words:** Cattle Drinking Trough, Automatic Drinking Trough, Real Time System, Water Level Control, Turbidity Sensor.

## Índice

<b>Índice</b> .....	11
<b>Índice de Figuras</b> .....	15
<b>Índice de Tablas</b> .....	17
<b>Introducción</b> .....	18
<b>Capítulo I. Diseño teórico</b> .....	19
<b>1.1. Planteamiento de la Investigación Síntesis de la Situación Problemática</b> .....	19
<b>1.2. Formulación del Problema de Investigación</b> .....	21
<b>1.3. Justificación de la Investigación</b> .....	21
<b>1.4. Objetivos</b> .....	22
<b>1.4.1. Objetivo General</b> .....	22
<b>1.4.2. Objetivos Específicos</b> .....	22
<b>1.5. Hipótesis</b> .....	22
<b>1.6. Antecedentes</b> .....	22
<b>1.6.1. Internacionales</b> .....	22
<b>1.6.2. Nacionales</b> .....	24
<b>1.6.3. Locales</b> .....	26
<b>1.7. Bases Teóricas</b> .....	26
<b>1.7.1. Prototipo</b> .....	26
<b>1.7.2. Sistema en Tiempo Real</b> .....	27
<b>Requisitos Temporales</b> .....	27
<b>Aplicaciones</b> .....	27
<b>1.7.3. Metodología para implementar sistemas en tiempo real</b> .....	28
<b>Metodología utilizada en el desarrollo del Prototipo</b> .....	28
<b>1. Comunicación</b> .....	28
<b>2. Plan Rápido</b> .....	28
<b>3. Modelado de Diseño Rápido</b> .....	28
<b>4. Construcción del Prototipo</b> .....	29
<b>5. Despliegue, Entrega y Retroalimentación</b> .....	29
<b>Metodología utilizada para la construcción del Prototipo</b> .....	29
<b>1.7.4. Bebedero de Ganado Vacuno</b> .....	30
<b>1.7.5. Medición del Nivel de Agua</b> .....	30

Interrupor de Flotador.....	30
1.7.6. <i>Software de Programación IDE Arduino</i> .....	30
1.7.7. <i>App Blynk</i> .....	31
1.7.8. <i>Fritzing</i> .....	31
1.7.9. <i>Sensor de nivel de líquido horizontal</i> .....	31
1.7.10. <i>Placa ESP32</i> .....	32
1.7.11. <i>Válvula solenoide</i> .....	32
1.7.12. <i>Fuente de alimentación</i> .....	32
1.7.13. <i>Módulo Relé</i> .....	33
1.7.14. <i>Sensor de turbidez de agua</i> .....	33
1.7.15. <i>Sensor Ultrasónico</i> .....	33
1.7.16. <i>Interrupor de botón de enclavamiento</i> .....	33
Capitulo II. Métodos y Materiales.....	34
2.1. Definición y Operacionalización de Variables.....	34
2.1.1. <i>Definición de Variables</i> .....	34
2.2. Tipo de Investigación .....	34
2.3. Diseño de contrastación de hipótesis/procedimiento a seguir en la investigación .....	35
2.4. Población, muestra.....	35
2.4.1. <i>Población</i> .....	35
2.4.2. <i>Muestra</i> .....	35
2.5. Métodos Aplicados en la Investigación .....	36
2.5.1. <i>Métodos para la búsqueda de información</i> .....	36
Histórico – Lógico. ....	36
Sistematización.....	36
2.5.2. <i>Métodos para construcción del conocimiento</i> .....	36
La modelación. ....	36
2.6. Técnicas, instrumentos, equipos y materiales.....	36
2.6.1. <i>Técnicas</i> .....	37
Observación sistemática. ....	37
Entrevista.....	37
2.6.2. <i>Instrumentos</i> .....	37
Guía de Observación.....	37
Guía de Entrevista. ....	37

2.6.3.	<i>Equipos</i>	37
	Equipo Humano.	37
	Equipo Tecnológico.	38
2.6.4.	<i>Materiales</i>	38
	Sensor de turbidez de agua.	38
	Sensor de nivel de líquido horizontal.	38
	Placa ESP32.	39
	Válvula solenoide ½" 12 VDC.	40
	Fuente de alimentación DC 12V 1A.	40
	Fuente de alimentación DC 5V 1A	41
	Módulo Relé 1CH 5VDC.	42
	Sensor Ultrasónico.	42
	Interruptor de botón de enclavamiento.	43
Capítulo III. Desarrollo de la Propuesta.		44
3.1.	<i>Comunicación.</i>	44
3.1.1.	<i>Aplicación de instrumentos para la recolección de datos</i>	44
3.1.2.	<i>Análisis de requisitos</i>	44
	Requisitos Funcionales.	44
	Caso de Uso Llenar bebedero.	45
	Caso de Uso Detener llenado cuando el agua este sucia.	45
	Caso de Uso Verificar nivel de agua actual.	46
	Caso de Uso Detener llenado por nivel máximo.	47
	Requisitos No Funcionales.	48
3.2.	<i>Plan Rápido</i>	50
3.2.1.	<i>Análisis de costos y presupuestos</i>	50
3.2.2.	<i>Calendarización del prototipo</i>	52
3.2.3.	<i>Análisis de factibilidad</i>	53
	Justificación técnica y económica	53
	Justificación tecnológica	54
3.3.	<i>Modelado y diseño rápido</i>	54
3.3.1.	<i>Diseño lógico</i>	54
3.3.2.	<i>Diseño físico</i>	55
3.3.3.	<i>Diseño del sistema de información (diagramas de casos de uso, secuencia o actividad)</i>	55

<b>Modelo de Casos de Uso del Sistema</b> .....	55
<b>Diagramas de Casos de Uso del Sistema.</b> .....	56
• <i>CUS01: Llenar bebedero.</i> .....	56
• <i>CUS02: Detener llenado cuando el agua esté sucia.</i> .....	56
• <i>CUS03: Verificar nivel de agua actual.</i> .....	57
• <i>CUS04: Detener llenado por nivel máximo.</i> .....	57
<b>Diagramas de Actividad del Sistema.</b> .....	58
• <i>Diagrama de Actividades 01: Llenar bebedero.</i> .....	58
• <i>Diagrama de Actividades 02: Detener llenado cuando el agua esté sucia</i> .....	59
• <i>Diagrama de Actividades 03: Verificar nivel de agua actual</i> .....	60
• <i>Diagrama de Actividades 04: Detener llenado por nivel máximo</i> .....	61
<b>Diagrama de Secuencia.</b> .....	62
• <i>Diagrama de Secuencia 01: Llenar bebedero.</i> .....	62
• <i>Diagrama de Secuencia 02: Detener llenado cuando el agua esté sucia</i> .....	63
• <i>Diagrama de Secuencia 03: Verificar nivel de agua actual</i> .....	63
• <i>Diagrama de Secuencia 04: Detener llenado por nivel máximo</i> .....	64
<b>3.4. Construcción del prototipo.</b> .....	64
3.4.1. <i>Testeo de materiales</i> .....	64
3.4.1.1. <i>Válvula Solenoide y Relé</i> .....	64
3.4.1.2. <i>Sensor de Nivel de Líquido Horizontal</i> .....	66
3.4.1.3. <i>Sensor de Turbidez de Agua</i> .....	69
3.4.1.4. <i>Sensor Ultrasónico</i> .....	73
3.4.2. <i>Conexión de la App Blynk y el sistema</i> .....	75
3.4.3. <i>Codificación del sistema</i> .....	90
3.4.4. <i>Construcción del bebedero a escala.</i> .....	101
3.4.5. <i>Ensamblaje o armado de prototipo</i> .....	102
<b>3.5. Despliegue entrega y retroalimentación.</b> .....	103
3.5.1. <i>Pruebas de Funcionamiento</i> .....	103
3.5.1.1. <i>Llenado de Bebedero</i> .....	103
3.5.1.2. <i>Cuando el agua está limpia</i> .....	104
3.5.1.3. <i>Cuando el agua está turbia</i> .....	105
3.5.1.4. <i>Vaciado de bebedero</i> .....	106
3.5.2. <i>Entrega</i> .....	107

<b>Capítulo IV. Resultados y Discusión.</b>	108
<b>Capítulo V. Conclusiones</b>	110
<b>Capítulo VI. Recomendaciones.</b>	112
<b>Bibliografía</b>	113
<b>Anexos.</b>	118
<b>Anexo N°1</b>	118
<b>Anexo N°2</b>	119
<b>Anexo N°3</b>	120
<b>Anexo N°4</b>	121
<b>Anexo N°5</b>	122
<b>Anexo N°6</b>	124

### Índice de Figuras

<b>Figura 1</b> <i>El paradigma de hacer prototipos.</i>	29
<b>Figura 2</b> <i>Sensor de turbidez de agua.</i>	38
<b>Figura 3</b> <i>Sensor de nivel de líquido horizontal.</i>	39
<b>Figura 4</b> <i>NODEMCU-32 30-PIN ESP32 WIFI.</i>	39
<b>Figura 5</b> <i>Válvula solenoide ½" 12 VDC.</i>	40
<b>Figura 6</b> <i>Fuente de alimentación DC 12V 1A.</i>	41
<b>Figura 7</b> <i>Fuente de alimentación DC 5V 1A.</i>	41
<b>Figura 8</b> <i>Módulo Relé 1CH 5VDC.</i>	42
<b>Figura 9</b> <i>Sensor Ultrasónico.</i>	42
<b>Figura 10</b> <i>Interruptor de botón de enclavamiento.</i>	43
<b>Figura 11</b> <i>Calendarización del prototipo.</i>	52
<b>Figura 12</b> <i>Diagrama de Bloques.</i>	54
<b>Figura 13</b> <i>Diagrama de Componentes.</i>	55
<b>Figura 14</b> <i>Modelo de Casos de Uso del Sistema (modelo o modelado).</i>	56
<b>Figura 15</b> <i>CUS01: Llenar bebedero.</i>	56
<b>Figura 16</b> <i>CUS02: Detener llenado cuando el agua esté sucia.</i>	56
<b>Figura 17</b> <i>CUS03: Verificar nivel de agua actual.</i>	57
<b>Figura 18</b> <i>CUS04: Detener llenado por nivel máximo.</i>	57
<b>Figura 19</b> <i>Diagrama de Actividades 01: Llenar bebedero.</i>	58
<b>Figura 20</b> <i>Diagrama de Actividades 02: Detener llenado cuando el agua esté sucia.</i>	59
<b>Figura 21</b> <i>Diagrama de Actividades 03: Verificar nivel de agua actual.</i>	60
<b>Figura 22</b> <i>Diagrama de Actividades 04: Detener llenado por nivel máximo.</i>	61
<b>Figura 23</b> <i>Diagrama de Secuencia 01: Llenar bebedero.</i>	62
<b>Figura 24</b> <i>Diagrama de Secuencia 02: Detener llenado cuando el agua esté sucia.</i>	63
<b>Figura 25</b> <i>Diagrama de Secuencia 03: Verificar nivel de agua actual.</i>	63
<b>Figura 26</b> <i>Diagrama de Secuencia 04: Detener llenado por nivel máximo.</i>	64
<b>Figura 27</b> <i>Testeo del control del módulo Relé mediante señal digital en la placa ESP32.</i>	65

<b>Figura 28</b>	<i>Relé Activado</i> .....	65
<b>Figura 29</b>	<i>Relé Desactivado</i> .....	66
<b>Figura 30</b>	<i>Test del control del sensor de líquido horizontal con la placa ESP32</i> .....	66
<b>Figura 31</b>	<i>Monitor Serie "Tanque vacío".</i> .....	67
<b>Figura 32</b>	<i>Monitor Serie "Tanque lleno".</i> .....	68
<b>Figura 33</b>	<i>Sensor de nivel activado</i> .....	68
<b>Figura 34</b>	<i>Sensor de nivel desactivado</i> .....	69
<b>Figura 35</b>	<i>Test del control del sensor de turbidez de agua con la placa ESP32</i> .....	71
<b>Figura 36</b>	<i>Sensor de Turbidez conectado a la placa ESP32, sumergido en agua limpia</i> .....	72
<b>Figura 37</b>	<i>Sensor de Turbidez conectado a la placa ESP32, sumergido en agua turbia</i> .....	72
<b>Figura 38</b>	<i>Sensor Ultrasónico conectado a la placa ESP32, parte 1</i> .....	74
<b>Figura 39</b>	<i>Sensor Ultrasónico conectado a la placa ESP32, parte 2</i> .....	74
<b>Figura 40</b>	<i>Ventana principal de la página de App Blynk.</i> .....	75
<b>Figura 41</b>	<i>Ventana para crear una cuenta de la App Blynk</i> .....	76
<b>Figura 42</b>	<i>Correo de activación</i> .....	76
<b>Figura 43</b>	<i>Creación de contraseña de Blynk.</i> .....	77
<b>Figura 44</b>	<i>Ventana de Blynk.</i> .....	77
<b>Figura 45</b>	<i>Creación de plantillas de Blynk, parte 1</i> .....	78
<b>Figura 46</b>	<i>Creación de plantillas de Blynk, parte 2</i> .....	78
<b>Figura 47</b>	<i>Guardar plantilla.</i> .....	79
<b>Figura 48</b>	<i>Crear dispositivo</i> .....	79
<b>Figura 49</b>	<i>Elección de dispositivos.</i> .....	80
<b>Figura 50</b>	<i>Creación de un nuevo dispositivo</i> .....	80
<b>Figura 51</b>	<i>Edición de panel.</i> .....	81
<b>Figura 52</b>	<i>Flujo de datos</i> .....	82
<b>Figura 53</b>	<i>Creación de la variable Turbidez</i> .....	82
<b>Figura 54</b>	<i>Variable Turbidez.</i> .....	83
<b>Figura 55</b>	<i>Creación de la variable NivelBebedero.</i> .....	83
<b>Figura 56</b>	<i>Creación de la variable VálvulaSuperior</i> .....	84
<b>Figura 57</b>	<i>Creación de la variable VálvulaInferior</i> .....	84
<b>Figura 58</b>	<i>Variables Creadas en la Datastream</i> .....	85
<b>Figura 59</b>	<i>Descarga de la App Blynk</i> .....	85
<b>Figura 60</b>	<i>Inicia sesión de la App Blynk</i> .....	86
<b>Figura 61</b>	<i>Logueo de la App Blynk</i> .....	86
<b>Figura 62</b>	<i>Dispositivo creado en la web de Blynk</i> .....	87
<b>Figura 63</b>	<i>Ventana en modo desarrollo de la App Blynk</i> .....	87
<b>Figura 64</b>	<i>Creación de botones en la App Blynk</i> .....	88
<b>Figura 65</b>	<i>Interfaz de Usuario</i> .....	89
<b>Figura 66</b>	<i>Credenciales de Blynk</i> .....	90
<b>Figura 67</b>	<i>Inclusión de la biblioteca ESP_Mail_Client</i> .....	90
<b>Figura 68</b>	<i>Definición de constantes para el envío de correo.</i> .....	91
<b>Figura 69</b>	<i>Creación de función para envío de correos.</i> .....	91
<b>Figura 70</b>	<i>Definición de constantes de los pines.</i> .....	92
<b>Figura 71</b>	<i>Declaración de Variables</i> .....	92



<b>Figura 72</b> <i>Declaración de pines.</i> .....	93
<b>Figura 73</b> <i>Funciones e Intervalos de tiempo.</i> .....	93
<b>Figura 74</b> <i>Lectura de variables</i> .....	94
<b>Figura 75</b> <i>Función sendWaterLevel</i> .....	94
<b>Figura 76</b> <i>Función sendTurbidity</i> .....	95
<b>Figura 77</b> <i>Función sendValveState</i> .....	96
<b>Figura 78</b> <i>Lectura del estado del botón de desagüe.</i> .....	96
<b>Figura 79</b> <i>Función valveControl</i> .....	97
<b>Figura 80</b> <i>Función measureTurbidity</i> .....	98
<b>Figura 81</b> <i>Función measureWaterLevel</i> .....	99
<b>Figura 82</b> <i>Función mapFloat</i> .....	99
<b>Figura 83</b> <i>Función sendMail</i> .....	100
<b>Figura 84</b> <i>Bebedero escala</i> .....	102
<b>Figura 85</b> <i>Operacionalización de variables</i> .....	118
<b>Figura 86</b> <i>Guía de observación para realizar el prototipo de sistema en tiempo real</i> .....	119
<b>Figura 87</b> <i>Diseño de la placa</i> .....	122
<b>Figura 88</b> <i>Soldado de borneras y header de pines</i> .....	122
<b>Figura 89</b> <i>Sensores conectados a la placa</i> .....	123
<b>Figura 90</b> <i>Respuesta de la encuesta para evaluar el correcto funcionamiento del proyecto</i> .....	124

### Índice de Tablas

<b>Tabla 1</b> <i>Requisitos Funcionales del Sistema</i> .....	44
<b>Tabla 2</b> <i>Caso de Uso Llenar Bebedero.</i> .....	45
<b>Tabla 3</b> <i>Caso de Uso Detener Llenado Cuando el Agua esté Sucia</i> .....	46
<b>Tabla 4</b> <i>Caso de Uso Verificar Nivel de Agua Actual</i> .....	47
<b>Tabla 5</b> <i>Caso de Uso Detener Llenado por nivel máximo.</i> .....	48
<b>Tabla 6</b> <i>Requisitos No Funcionales del Sistema</i> .....	49
<b>Tabla 7</b> <i>Materiales usados en el proyecto</i> .....	50
<b>Tabla 8</b> <i>Costos del hardware.</i> .....	51
<b>Tabla 9</b> <i>Servicios.</i> .....	51
<b>Tabla 10</b> <i>Costos totales.</i> .....	51
<b>Tabla 11</b> <i>Justificación técnica y económica</i> .....	53
<b>Tabla 12</b> <i>Funcionamiento del llenado de bebedero</i> .....	103
<b>Tabla 13</b> <i>Funcionamiento del bebedero cuando el agua está limpia</i> .....	104
<b>Tabla 14</b> <i>Funcionamiento del bebedero cuando el agua está turbia.</i> .....	105
<b>Tabla 15</b> <i>Funcionamiento del vaciado del bebedero</i> .....	106
<b>Tabla 16</b> <i>Evaluación del funcionamiento del sistema en tiempo real</i> .....	107

## **Introducción**

El presente documento de investigación titulado Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para Controlar Nivel de Agua en la Quinta Fernández, tiene como objetivo implementar un prototipo de sistema en tiempo real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández.

El sistema ha sido desarrollado con el fin de tener un mejor control del llenado de bebederos para el ganado vacuno, ya que muchas veces los bebederos se quedan sin agua, y esto genera problemas de productividad a los ganaderos, y en muchos casos incluso puede provocar la muerte del ganado; otro de los grandes problemas que se tiene es que muchas veces el agua se desperdicia y humedece el corral, afectando su productividad y la economía.

El tipo de investigación que usamos según su propósito en este caso, es la Investigación Aplicada Tecnológica, de acuerdo con su nivel de profundización será una Investigación Explicativa, dependiendo el tipo de datos utilizados, se aplicará la Investigación Cuantitativa, dependiendo del grado de manipulación de variables, es una investigación No Experimental, con Diseño Transeccional o Transversal, Correlacional-Causal, según el tipo de inferencia hemos considerado que es una Investigación Hipotética-deductiva y de acuerdo el tiempo en el que se realiza, la hemos clasificado como una Investigación Transversal.

Esta tesis se divide en seis capítulos, el primer capítulo se estableció el Diseño Teórico, donde se hace un análisis de la situación problemática del estado en las que se encuentran los bebederos de la quinta Fernández en cuanto a su control de llenado, además se describen los objetivos de este proyecto y se da una justificación de la investigación realizada. El segundo Capítulo se refirió a los Métodos y Materiales, donde se detalla e identifica métodos, técnicas y materiales que serán necesarios para el desarrollo del proyecto. El tercer Capítulo trata sobre el Desarrollo de la Propuesta, siguiendo los cinco pasos de la metodología Evolutiva por Prototipos, Comunicación, Plan rápido, Modelado Diseño rápido, Construcción del prototipo y como último paso Despliegue, entrega y retroalimentación. En el cuarto Capítulo se indicó los Resultados y Discusión, donde se muestra y analiza los resultados obtenidos. En el quinto Capítulo se nombraron las conclusiones. Y por último en el sexto Capítulo se aludieron las recomendaciones de este proyecto.

## **Capítulo I. Diseño teórico.**

### **1.1. Planteamiento de la Investigación Síntesis de la Situación Problemática**

El uso de bebederos contribuye a la salud del ganado. Evita que el ganado se deshidrate y proporciona el agua necesaria para apoyar las funciones fisiológicas del animal, al igual que la digestión y metabolismo de nutrientes.

Según Langer (2021), en Israel, los sistemas informáticos contribuyen a registrar datos sobre la leche y a gestionar la salud del ganado, las raciones de alimentos, la cría adecuada y una sólida planificación económica. A medida que avancemos, estos sistemas pueden volverse más eficaces y permitir una mejor planificación.

Según SENASA (n.d.) (Servicio Nacional de Sanidad y Calidad Agroalimentaria) de Argentina, para considerar el valor del consumo de agua del ganado lechero se debe seguir una serie de indicadores; el 80% del consumo diario es después del ordeño por lo que el ganado puede beber por minuto de 15 hasta 20 litros de agua, para producir un litro de leche, el 10 al 25% del lote o rodeo deben beber agua en simultáneo. Una vaca puede llegar a consumir agua entre 5 a 10 veces al día, bebiendo de 3 a 4 litros en corrales libres con bebederos accesibles.

Según Fernández Cirelli et al. (2017), en Argentina, para la producción ganadera el agua es fundamental ya que en rodeos lecheros de alto rendimiento, necesitan una alta disponibilidad de agua de bebida de buena calidad; pues se observó que la productividad de leche es superior cuando el agua está disponible de manera ininterrumpida que cuando se administra una sola vez al día, ya que en el primer caso beben más agua. Esto es particularmente notable en las vacas de alta producción, donde una reducción del 1,3 % en la ingesta normal de agua puede provocar una disminución del 7,5 % en la producción de leche.

El Instituto Nicaragüense de Tecnología Agropecuaria (INTA) & Instituto Nacional Tecnológico (INATEC) (2010), nos afirma que uno de los principales requisitos mínimos para el manejo del ganado es garantizar que haya agua fresca y limpia disponible en cantidades suficientes. Teniendo en cuenta que el agua constituye el 70% del cuerpo del ganado adulto y más del 90 % de los recién nacidos. Un animal muere, en el caso pierda la quinta parte del agua de su cuerpo. (p.10)

Además el Instituto Nicaragüense de Tecnología Agropecuaria (INTA) & Instituto Nacional Tecnológico (INATEC) (2010), nos dice que la cantidad consumida debe ser la deseada o por lo

menos 2 a 3 veces al día, debe ser fresca, libre de malos olores o sabores desagradables y sin sustancias nocivas. El suministro de agua debe estar limpio, en el caso que el agua esté sucia, tomará y comerá menos, lo que reduce la producción. (p.10)

La Agencia Agraria de Noticias, (2021), nos indica que con el objetivo de fortalecer la industria láctea nacional, el 19 de julio del 2021, fue aprobada por insistencia la Ley N°31296-Ley de promoción de la cadena productiva ganadera lechera por el Congreso de la República.

Según Diario Oficial El Peruano, (2021), dicha ley tiene como objetivo establecer medidas para promover el crecimiento de la cadena productiva ganadero-lechera con el fin de generar puestos de trabajo masivos en el campo, fortalecer los mercados, a las micro empresas y a las PIMES, que son el motor del desarrollo de la economía local, regional y nacional.

En algunos lugares, como en los distritos de Huancabamba, Oxapampa y Chontabamba, el mal manejo de la ganadería extensiva a menudo aumenta la contaminación de los ríos y quebradas que abastecen de agua a muchos centros poblados rurales y urbanos, poniendo en peligro la buena condición del agua consumida por miles de habitantes. Una de las opciones más útiles para moderar esta amenaza es instalar bebederos para el ganado vacuno. Instituto del Bien Común (2017)

En el sector rural La Clake del distrito de Reque, provincia de Chiclayo, departamento de Lambayeque, se encuentra ubicada La Quinta “Fernández”, la cual se dedica a la crianza de ganado vacuno; ya que a un futuro tiende a ser líder en distribución de productos lácteos en la región Lambayeque.

Para ello la quinta necesita llevar a cabo un estricto control sobre el llenado de bebederos.

La problemática que se tiene es del llenado de bebederos para el ganado vacuno, muchas veces los bebederos se quedan sin agua, y esto genera problemas de productividad a los ganaderos, y en muchos casos incluso puede provocar la muerte del ganado; si los animales no cuentan con el agua suficiente pueden padecer de deshidratación, una enfermedad que incluso causa muchas muertes de los animales; otro de los grandes problemas que se tiene es que muchas veces el agua se desperdicia, ya que el ganadero olvida los grifos abiertos, y en estos tiempos el agua es vital, y sabiendo que el uso responsable de los recursos hídricos es el principal reto al que se enfrenta la humanidad, debemos cuidar y reducir el gasto innecesario de agua, además que esto provoca

inundación en el corral del ganado vacuno, dando un problema más al ganadero, ya que el ganado tendrá que dormir en un lugar húmedo, y su rendimiento lechero no será el mismo, afectando su productividad y la economía.

Langer (2021), nos dice que durante los últimos 100 años, el perfeccionamiento vino en gran medida de los cruces innovadores que ha dado como resultado un ganado saludable y productivo. El crecimiento parece provenir del uso de la tecnología.

### **1.2. Formulación del Problema de Investigación**

¿El prototipo de sistema en tiempo real de bebederos de ganado vacuno, mejorará el control del nivel de agua en la Quinta Fernández?

### **1.3. Justificación de la Investigación**

Aumento de la producción lechera, destinada para la elaboración de derivados lácteos, los cuales contribuyen al sustento familiar, la seguridad alimentaria y la nutrición de los hogares, además del incremento de ganancias en la Quinta Fernández.

Justificación Económica, permitirá mantener a los animales hidratados las 24 horas del día para un mejor rendimiento en la producción lechera, además la disponibilidad de agua limpia, dando lugar a un ganado sano y productivo.

En base a los argumentos teóricos existentes se formulará un Sistema en Tiempo Real, los cuales son sistemas informáticos que por su naturaleza o función interactúan constantemente con el entorno externo que cambian dinámicamente en el tiempo físico, y también deben generar respuestas de tiempo limitado a los eventos que admite en su entorno, así mismo logran generar eventos y acciones al entorno en momentos particulares del tiempo los cuales contribuyan al desarrollo científico y teórico en esta especialidad.

La justificación práctica se define por la utilización de microcontroladores y circuitos electrónicos que pueden ser programados para realizar acciones o instrucciones que nosotros deseemos, en nuestro caso, para crear el Sistema en Tiempo Real.

Justificación social viene dada por que esta investigación podrá ser aplicada a cualquier Quinta donde se críe ganado vacuno, especialmente en ámbito rural donde el llenado de bebederos se da

de forma manual con desperdicios de agua, inundación de corrales y ausencia de personal para garantizar el abastecimiento de agua las 24 horas del día.

En la justificación metodológica se usará Metodologías de Desarrollo de Software, las cuales tienen como objetivo cumplir las necesidades del cliente que vienen dadas en forma de especificaciones, respetando los plazos y los costos, alcanzando determinados niveles de calidad.

## **1.4.Objetivos**

### ***1.4.1. Objetivo General***

Implementar un prototipo de Sistema en tiempo real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández.

### ***1.4.2. Objetivos Específicos***

Diagnosticar la problemática del control del nivel del agua en bebederos de ganado vacuno.

Diseñar el prototipo del sistema en tiempo real que se utilizará en la implementación del proyecto.

Codificar el sistema en tiempo real utilizando el lenguaje de programación y la plataforma Arduino e inteligencia artificial.

Realizar las pruebas correspondientes en el prototipo propuesto para garantizar su buen funcionamiento.

## **1.5.Hipótesis**

Con el prototipo de sistema en tiempo real de bebederos de ganado vacuno, se mejorará el control del nivel de agua en la Quinta Fernández.

## **1.6.Antecedentes**

### ***1.6.1. Internacionales***

Osuna & Calderone (2018) en su tesis “Monitoreo Planta de Agua mediante Sistema Arduino”, el proyecto tiene como finalidad elaborar un prototipo, para el monitoreo de la planta de tratamiento de agua de la Cooperativa de Agua Potable de la ciudad de Roldán, Santa Fe, apoyándose en la gestión remota con el Sistema Arduino. Se trata de la construcción de un dispositivo controlado por una placa Arduino que permite el monitoreo del sistema a través del internet, mediante un servidor alojado en la nube, donde se puede obtener en tiempo real el estado actual de la instalación en cuanto a tensión proveniente de un proveedor (Empresa Provincial de la Energía) y el estado

del generador de energía (Encendido/ apagado/automático). El sistema además puede enviar alertas por SMS a un número de telefonía móvil configurado con anticipación y/o correo electrónico a diversos buzones. El contenido de estos mensajes tiene como propósito orientar al personal que labora durante la respectiva instalación para que puedan proceder con la instalación a través de nuestros equipos, en caso de ser necesario.

Ordóñez Contreras & Ponce Jácome (2019) en su proyecto de titulación “Diseño y desarrollo de prototipo de un dispositivo de medición de la calidad del agua de pozos artesanales aplicado en el recinto el Cacique de la Parroquia Junquillal del Cantón Salitre”, tiene como fin diseñar un dispositivo el cual utilice sensores para medir la calidad del agua, examinar el pH, la turbidez y la conductividad en tiempo real para establecer si hay sustancias pesadas o partículas en suspensión que puedan contener trazas orgánicas inapropiadas para el consumo humano así como se muestran en el aplicativo móvil de Android.

Torres Stuar & Quintero Chocue (2019) en su tesis “Diseño e implementación de un dispensador remoto de alimento y agua para perros en la plataforma Arduino”, tiene como objetivo asegurar la alimentación de nuestros perros, a través de un dispositivo móvil que controla de forma remota un dispensador, suministrando agua y comida periódicamente según la necesidad. Tiene una capacidad de alimentación correspondiente a 3kg, el agua la proporciona la casa conectando el aparato a un grifo y será controlada por el usuario según la necesidad del bidón, estas especificaciones nos dan autonomía para 2 días de alimentación si el perro es de raza mediana. El proyecto se ejecuta bajo el modelo en cascada, garantizando un enfoque ordenado y secuencial, siguiendo con rigurosidad cada una de sus etapas. El dispositivo brinda total libertad y tranquilidad a los dueños del perro permitiéndoles alimentarse a distancia durante varios días, asegurando una adecuada nutrición y bienestar de la mascota con un producto de fácil uso.

Córdoba et al. (2020) en el artículo científico Guacamaya, encontramos el “Prototipo de sistema, utilizando arduino y gsm, para el monitoreo del flujo de agua potable en las tuberías, del IDAAN en la provincia de Coclé”, esta investigación se enfoca en diseñar y construir un prototipo con Arduino y la red GSM (Global System Mobile) con el propósito de recolectar datos del recorrido del agua y luego enviar notificaciones por SMS; este proyecto tiene dos partes: hardware y software. En cuanto al hardware se realizó un análisis de los diversos componentes con la intención

de tomar datos y hacer uso del sensor de flujo de agua, de donde se obtuvieron los valores del recorrido de agua que fluye por las tuberías en litros por minutos.

Quingatuña Tasigchana (2020) en su tesis “ Implementación de un prototipo para extracción de agua de un pozo mediante la medición de caudal y nivel del agua utilizando la plataforma Arduino y aplicación móvil Android”, tiene como finalidad optimar el tiempo y rehuir el trabajo manual humano, además esta investigación tiene como objetivo extraer el agua del pozo de forma automática midiendo el caudal y nivel del agua haciendo uso de la aplicación móvil Android y la plataforma Arduino, en el cual se procura conservar un nivel suficiente de agua en el tanque de almacenamiento y una circulación de agua suficiente para satisfacer las necesidades de las familias. Dicho prototipo se puede monitorear y controlar a través de la telefonía móvil a través de una conexión Bluetooth mediante una aplicación móvil elaborada en App Inventor y/o mediante SMS a través de la red celular. Asimismo, el prototipo cuenta con un sistema de alertas en el que indican diversos eventos que tiene el prototipo, como la variabilidad de su funcionamiento, nivel de agua, déficit de la capacidad, estado de obstrucción y la situación en la que se encuentra la tarjeta Micro SD.

### **1.6.2. Nacionales**

Depaz Sandon (2018) “Prototipo usando tecnología Arduino para medición de nivel de agua en lagunas peligrosas del parque Nacional Huascarán. 2018”, tiene como fin, desarrollar un prototipo utilizando tecnología Arduino con la intención de medir el nivel de agua en las lagunas peligrosas del Parque Nacional Huascarán (PNH), permitiendo monitorear y prevenir la variación de nivel de agua de las peligrosas lagunas del PNH. Dado que los niveles de agua deben ser monitoreado a diario por el personal del gobierno regional y las organizaciones responsables, por lo que nuestro prototipo seguirá la metodología sugerida por estas organizaciones para la recolección de datos y la posterior verificación con análisis de variación entre los datos de los trabajadores recopilados de manera manual y automática.

Ocola Gadea & Ramos Mendoza (2018) en su tesis “Implementación de una aplicación móvil complementada con la tecnología de sensores multiparamétricos con arduino para monitorear la calidad del agua potable en el distrito de Laredo, Trujillo.”, dicho proyecto tiene como principal objetivo la implementación de una aplicación móvil la cual permite monitorizar la calidad del agua en los pozos de los caseríos del distrito de Laredo, haciendo uso de un dispositivo electrónico



analizando los siguientes indicadores: Potencial de Hidrógeno (pH), temperatura y turbidez, por medio de la tecnología Arduino mega y sensores de acuerdo a los indicadores mencionados anteriormente. Dicha problemática se presenta en la actualidad en nuestra nación, donde las comunidades que presentan el servicio de agua potable no miden frecuentemente la pureza del agua, esto se debe al tiempo que requiere la recopilación de datos, ya que personal instruido debe trasladarse a los caseríos frecuentemente para tomar las muestras respectivas y después analizarlas.

Huayta Sucasaca & Suaña Humpire (2017) en su tesis “Diseño e Implementación de un Sistema Scada para el control de nivel de agua para uso domótico mediante redes industriales”, tiene como propósito desarrollar un sistema para controlar el nivel de líquido con el fin de crear modernos espacios de control y facilidad. Con esta investigación hemos mostrado técnicas para el uso del programa LabVIEW- Arduino y el sistema de comunicación IEEE 802, creando un nivel de registro mediante la programación en Labview, utilizando la interfaz LINX entre Labview y Arduino. Y así adquirir y procesar los datos, para ellos haremos uso de los siguientes materiales: computadora portátil, Arduino Ethernet, Arduino UNO, bomba de agua DENSO 12V, fuente de 12V, sensor ultrasónico HC-SR04, una placa de relé, y el software Labview. Así mismo se elaboró un modelo para llevar a cabo las pruebas de operatividad, dando los resultados previstos.

Chavarry Vega & Acedo Sosa (2020) en su tesis “Evaluación Energética de un Sistema de Bombeo de Líquidos Automatizado con Arduino en la empresa ARPE EIRL”, propone un sistema de gestión con el fin de estimar el procedimiento de un sistema de bombeo de líquidos utilizando herramientas Arduino. El conjunto de métodos de investigación se ajusta al diseño experimental sistémico, con información de entrada y salida para la programación lógico - matemático. Se realizó el prototipo correspondiente con un sistema de bombeo usando dos bombas centrífugas que transfieren agua entre tres tanques de almacenamiento. Los indicadores para evaluar el consumo energético del sistema son la capacidad de almacenamiento y el tiempo de operatividad de los dispositivos, como la evaluación del funcionamiento permite determinar el consumo de energía mínimo, menor a 7 amperios, que se logra durante la operación con un volumen de  $0,32 \text{ m}^3$  de líquido conducido por la operación bach.

Jara Neyra et al. (2019) en su tesis “Prototipo de sistema de monitoreo y control de consumo de agua potable aplicando tecnología GSM”, está fundamentado en una línea de investigación del Internet de las Cosas y nace tras la alta demanda de agua potable, debido al crecimiento

demográfico y al derroche de este recurso, se destaca la importancia del agua monitoreando y controlando el consumo del recurso fundamental de nuestra vida diaria, debemos seguir siendo conscientes de este hecho porque en el Perú se pierde el 37% del agua cada año, según indicó el Jefe de la Autoridad Nacional del Agua (ANA). Apoyándonos en los datos del ANA, el derroche de agua es enorme, especialmente en nuestros hogares, con las fugas de agua, dejando los grifos abiertos, etc. Es por ello que para contribuir al mantenimiento de este recurso se ha desarrollado un prototipo de seguimiento y verificación del agua utilizando las Comunicaciones Móviles (GSM), que permite monitorear el uso de agua en  $m^3$ , y admitir el fluido utilizando la válvula solenoide mediante un SMS que posibilita el fluido del agua manejando su apertura y su cierre, los materiales utilizados son, una tarjeta Arduino, un relé, una válvula solenoide, un módulo GSM y un sensor de fluido de agua, este último es un sensor de tipo hélice que permite medir la cantidad de agua y estar al tanto de la lectura del consumo. Este trabajo se llevó a cabo con la finalidad de evitar su derroche, y para controlar el consumo de agua, contribuyendo al mantenimiento de este recurso muy importante en nuestra vida cotidiana.

### **1.6.3. Locales**

Quezada Diaz & Salazar Santamaria (2018) en su tesis “Diseño de un Sistema de Control de presión de agua en función de la demanda para el laboratorio de Ingeniería Electrónica - UNPRG”, tiene como finalidad diseñar un sistema de presión constante en la Escuela de Ingeniería Electrónica – UNPRG, cuya investigación consistió en desarrollar el número y patrón en cada instrumento sanitario indicar la presión, determinar la capacidad y velocidad que debe enviar el convertidor de velocidad, establecer la posición de la bomba, diámetro y la longitud de la canalización, así mismo la capacidad del tanque de almacenamiento. A continuación, se debe calcular la capacidad y el descenso de presión en cañerías y así definir la serie de las válvulas y la frecuencia de funcionamiento del convertidor de velocidad. Dicho programa debe estar de manera espontánea para que la electrobomba bombee agua con la presión respectiva según la longitud referente a la tubería y la disposición de equipos sanitarios.

## **1.7.Bases Teóricas**

### **1.7.1. Prototipo**

Pereira (2019) afirma que es un modelo a escala de lo real, pero no es lo suficientemente funcional como para ser equivalente al producto final, porque no realiza todas las tareas fundamentales del sistema final, proporcionando una rápida retroalimentación del usuario sobre el sistema, como se

citó en la tesis “Análisis y diseño de un prototipo de sistema de inventario de equipos informáticos de la empresa Frontera Energy - 2019” de Oversluijs Rengifo & Rodríguez Najar (2019)

### ***1.7.2. Sistema en Tiempo Real***

El Departamento de Arquitectura y Tecnología de Computadores (n.d.), define que un Sistema en Tiempo Real es aquel que debe proporcionar las respuestas apropiadas en un tiempo específico. Si el tiempo de respuesta supera ese límite, se producirá un descenso del rendimiento y/o fallas. El intervalo es aceptable si el tiempo entre la entrada y salida es adecuado.

Se clasifica de la siguiente manera:

#### **Requisitos Temporales.**

- Tiempo real estricto (hard real time):

Es imprescindible que la respuesta se produzca dentro de ciertos límites. Como ejemplo tenemos el control de vuelo.

- Tiempo real no estricto (soft real time):

En el caso que se consienta la pérdida temporal de especificaciones, esto debería ejecutarse como de costumbre. Este tipo de Sistema en Tiempo Real se utiliza principalmente para sistemas de recopilación de datos.

- Tiempo real firme (firm real time):

Esto ocurre cuando se permite una pérdida poco frecuente de especificaciones temporales, pero esta pérdida no implica provecho porque las respuestas aplazadas son eliminadas. Un buen ejemplo es el sistema multimedia.

#### **Aplicaciones.**

- **Control de procesos industriales:**

La tarea de la computadora es producir señales para lograr su propósito, dependiendo de la medición de la variable a manipular, es el valor que le se asigna para un cierto algoritmo de control. En definitiva, lo que se quiere es una variable que siga un determinado desarrollo como la temperatura, el caudal, la presión, etc

- **Manufactura:**

El computador es el encargado de administrar las tareas que ejecutaran los diversos elementos del sistema, al igual que las máquinas, herramientas, etc. El cual es posible controlar los procedimientos de producción con la finalidad de reducir de costes y/o asegurar la calidad.

- **Comunicación, mando y control:**

Son responsables de recopilar y mantener la información como apoyo a la toma de decisiones, estos Sistemas en Tiempo Real son usados en reserva de billetes, seguimiento de pacientes, control de la aviación, y más.

### ***1.7.3. Metodología para implementar sistemas en tiempo real***

Para el desarrollo de la propuesta se utilizarán dos metodologías, la primera para el desarrollo de software utilizando la metodología Evolutiva por prototipos y la segunda para la construcción del prototipo (hardware), aplicando la metodología de diseño Bottom-Up.

**Metodología utilizada en el desarrollo del Prototipo.** La metodología Evolutiva por Prototipos, según Pressman, (n.d.), el conjunto de objetivos generales para el software por lo general son definidos por el cliente, sin embargo no especifica requisitos exactos de las características y su funcionalidad. En otras ocasiones, el desarrollador puede no estar confiado de la eficiencia del algoritmo, y de la flexibilidad del sistema operativo o el tipo de interacción humano - máquina. En otras situaciones, las reglas de hacer prototipos pueden proporcionar la estrategia más apropiada.

Según Albarrán Trujillo (2016) y Borjas Martínez et al. (2017), la metodología Evolutiva por Prototipos tiene las siguientes fases:

**1. Comunicación.** Esta es la primera fase, aquí se define con el cliente los objetivos y requerimientos del software.

**2. Plan Rápido.** En esta fase se estiman los recursos y se planifican los tiempos para realizar el prototipo.

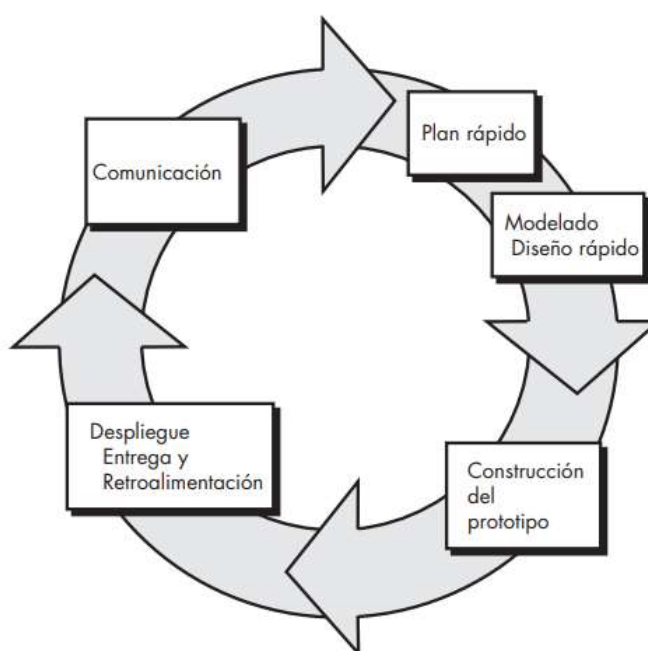
**3. Modelado de Diseño Rápido.** Aquí se realiza el modelado, centrándose en los aspectos del software que serán visibles para los usuarios finales, además se determina la estructura requerida para el software.

**4. Construcción del Prototipo.** En esta fase se lleva a cabo la programación y construcción del prototipo, que posteriormente junto al cliente será evaluado para poder refinar los requisitos del sistema que se desarrolló.

**5. Despliegue, Entrega y Retroalimentación.** Esta es la última fase, donde se hace entrega al usuario el sistema para ser puesto en práctica y verifique su funcionamiento.

**Figura 1**

*El paradigma de hacer prototipos.*



**Nota.** El modelo a seguir según la metodología Evolutiva por Prototipos. Tomada de (Pressman, n.d.).

**Metodología utilizada para la construcción del Prototipo.** Según Schweers, (2002), La metodología de diseño Bottom-Up no se refiere a una organización de componentes del sistema. Para establecer el diseño general, solamente se agrupa a los elementos de nivel bajo. Se debe empezar hacer una relación con esquemas de los componentes del circuito, para realizar un diseño Bottom-Up. En otros términos, los componentes generalmente se crean tomando en cuenta que forman parte de una biblioteca que incluyen elementos esenciales, que constituyen unidades de

funcionalidad con sus propios significados de diseño. Estas unidades se denominan primitivas, porque no se requieren elementos de nivel inferior con el fin de realizar el diseño.

#### ***1.7.4. Bebedero de Ganado Vacuno***

Según COLPOS (2016), un bebedero de ganado vacuno, es una estructura que consiste en un tanque rectangular o cuadrangular construido con piedra, armex, arena, cemento, ladrillo y grava, con el propósito de almacenar agua para abastecer al ganado, evitando el trabajo de los productores para transportar agua para el ganado, además reduce la distancia que recorren los animales para consumir agua, disminuyendo la pérdida de peso del ganado vacuno. Además, permite disponer siempre de agua en los bebederos, estos deben tener sombra para que el agua se mantenga fresca.

#### ***1.7.5. Medición del Nivel de Agua***

Según lo publicado en la Revista ElectroIndustria (2019), la medición de niveles de agua es una tarea indispensable en diversos procesos industriales. Teniendo en cuenta que los líquidos pueden tener propiedades diferentes, existen distintos métodos y dispositivos para llevar a cabo esta actividad, también nos dice que, para medir los niveles de agua, podemos referirnos a tres métodos principales de medición: directo, de presión hidrostática y de las características eléctricas del líquido.

El método de medición que usaremos es directo, ya que para nivel de líquidos se estudiarán los indicadores visuales y el instrumento de flotador.

**Interruptor de Flotador.** Según OMEGA (n.d.), es un sensor de nivel de punto, un flotador magnético se mueve sobre la superficie del líquido, activando un sellado herméticamente "reed switch, interruptor de láminas" en el tallo. Con el sencillo manejo se facilita la instalación, reduce el efecto, la vibración y la presión. El interruptor de láminas puede ser, de un solo polo, o de doble tiro.

#### ***1.7.6. Software de Programación IDE Arduino***

Arduino.cc (2020), nos dice que “el IDE Arduino, es un software de desarrollo de código abierto, que facilita la escritura del código y carga de cualquier placa Arduino.”; y la página web Arduino.cl (n.d.) nos dice que es un grupo de herramientas de software que facilita a los programadores desarrollar y documentar todo el código primordial, con la finalidad que Arduino se ejecute como

deseamos. El IDE de Arduino posibilita escribir, editar, depurar y guardar nuestro programa (conocidos como “sketches” en el mundo de Arduino) de una forma fácil.

Según Montenegro Ortiz (2016) el software de programación de Arduino (IDE) es compatible con las plataformas PC, Mac y Linux, además se puede descargar desde el sitio web ([www.arduino.cc](http://www.arduino.cc)). En esta página, encontrará instrucciones paso a paso para descargar e instalar el programa en la plataforma deseada. Una vez completada la instalación, se procede a conectar la placa Arduino a un ordenador a través de un cable USB. Como cualquier dispositivo USB, se conecta a una computadora, luego se procede a instalar los drivers respectivos, tal y como se detalla en la guía de la página web oficial de Arduino.

#### ***1.7.7. App Blynk***

Según Guaranga Allauca (2018), en su tesis nos indica que App Blynk es una plataforma de IoT, que permite combinar software y hardware, conectar dispositivos a la nube, diseñar aplicaciones para supervisar y controlar de manera remota de una manera sencilla y rápida.

Blynk está diseñado pensando en la programación interactiva, con variables virtuales y cajas (widgets), lo que facilita mucho el desarrollo de aplicaciones, en proyectos básicos no se requiere de programación adicional en el IDE de la placa controladora, basta con un amplio catálogo de widgets con la capacidad de leer entradas analógicas, leer de entradas digitales, pulsadores, displays, control de los periféricos de teléfonos móviles y más.

#### ***1.7.8. Fritzing***

Según la Oficina de Software y Hardware Libre (2014), este es un programa totalmente gratuito y de código abierto que habilita crear y diseñar circuitos. Tiene una interfaz sencilla e intuitiva, completadas con otras innovaciones como Arduino y Processing. Es un software de referencia de diseño electrónico gratuito, que permite a los diseñadores, investigadores, artistas y aficionados a diseñar sus propios prototipos basados en Arduino y formar esquemas de circuitos impresos para posteriormente ser fabricados.

#### ***1.7.9. Sensor de nivel de líquido horizontal***

También conocido como sensor de nivel, interruptor flotador o boya de nivel, dicho sensor presenta aislante eléctrico, ayudando a evitar los cortos circuitos. Es de fácil incorporación con

sistemas digitales como lo es Arduino, PIC o PLCs. Además, se comporta de forma similar a un interruptor o switch. Su funcionalidad principal nos permite conocer si el agua pasa cierto nivel.

#### ***1.7.10. Placa ESP32***

Según Ciro Edgardo & Alejandro, (2022), en su tesis nos indica que fue creada por la empresa Espressif Systems, esta placa es un factor clave para inspirar a la investigación y elaboración de proyectos sobre la integración de sistemas IoT, ya que es una placa fácil de adquirir en el mercado y a un bajo costo.

Además, se ajusta a una amplia variedad de implementaciones debido a su bajo consumo de energía, sus diversos entornos de código abierto y sus bibliotecas. Así mismo, posee una extensa documentación y varias comunidades activas con ejemplos de programación en lenguaje C/C++, Python y más.

#### ***1.7.11. Válvula solenoide***

Según Muñoz Llamuca & Torres Torres (2017), nos dice que es un tipo de instrumento que trabaja con corriente eléctrica, se usa para controlar el derrame de líquidos o gases, es decir en posición totalmente cerrada o abierta. El cierre se aplica por gravedad, presión o ejerciendo movimiento a un resorte, y se abre durante desplazamiento de un pistón que funciona bajo las cualidades magnéticas de una bobina que se halla energizada eléctricamente. Tener presente que la válvula solenoide no regula el fluido, aun cuando pueda estar completamente abierta o cerrada. (p.19).

#### ***1.7.12. Fuente de alimentación***

Según López (2020), es una parte importante de un dispositivo electrónico ya que es responsable de darle vida. En todo ordenador, por más pequeño que sea, siempre hay una fuente de alimentación, aunque no podamos verla. Desde Smartphones, hasta televisores y las computadoras, todos tienen un componente que se encarga de realizar la función de fuente de alimentación, es decir, administrar la fuente de alimentación de la red y adaptarla para proporcionar energía al dispositivo. Como tal, la fuente de alimentación de un dispositivo se encarga de proporcionar la corriente correcta y necesaria a los dispositivos electrónicos.



### ***1.7.13. Módulo Relé***

Robots Argentina (2020), nos dice que “Es un interruptor mecánico operado eléctricamente que se puede encender o apagar, dejando pasar la corriente o no, y se puede controlar con voltajes bajos, como los 5V que se usan para la alimentación de un Arduino”.

### ***1.7.14. Sensor de turbidez de agua***

La página web Naylamp Mechatronics (n.d.-b) nos indica que el sensor de turbidez de agua se utiliza en aguas residuales, alimentos y bebidas para medir la calidad del agua fácilmente. Los diodos infrarrojos y los fototransistores se colocan uno frente al otro para calcular la separación y transmisión de luz, detectando así diferentes partículas suspendidas en agua o TSS (sólidos totales en suspensión).

Naylamp Mechatronics (n.d.-b) también nos dice que el sensor tiene salidas analógicas y digitales, que varían con la turbidez del agua, lo que le permite distinguir rápidamente entre agua limpia y turbia o sucia. Para medir la sensibilidad de la salida digital, haciendo uso del módulo, se integra un potenciómetro. Además, el sensor de turbidez, puede ser usado para calibrar la condición del agua en tanques de agua, riachuelos, piscinas y más.

### ***1.7.15. Sensor Ultrasónico***

Naylamp Mechatronics, (n.d.-h) nos dice que es un sensor que se caracteriza por su pequeño tamaño, bajo consumo energético y buena precisión, es considerado un sensor de distancia que hace uso de ultrasonido (sonar) para precisar la distancia de un objeto en un rango de 2 a 450 cm.

### ***1.7.16. Interruptor de botón de enclavamiento***

Naylamp Mechatronics (n.d.-d) nos indica que se utiliza como un mecanismo de protección, es decir para evitar circunstancias que pueden dañar los equipos o puedan poner en peligro vidas humanas, además nos permite ahorrar energía y posibles gastos ante algún desperfecto. Su función basta con tocar el botón para activar el mecanismo inmediatamente.

## **Capítulo II. Métodos y Materiales.**

### **2.1. Definición y Operacionalización de Variables**

#### **2.1.1. Definición de Variables**

- Variable Independiente (X)

Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno.

- Variable Dependiente (Y)

Nivel de Agua en la Quinta Fernández.

### **2.2. Tipo de Investigación**

El tipo de investigación, de acuerdo con su propósito en este caso, es la Investigación Aplicada Tecnológica porque se nutre de la teoría para generar conocimiento práctico, y es muy común utilizarlas en ramas del conocimiento como la ingeniería, con la finalidad de impulsar un impacto positivo en la cotidianidad.

Según su nivel de profundización será una Investigación Explicativa ya que es la investigación más común y encargada de establecer relaciones de causa y efecto que posibilita hacer generalizaciones que puedan extenderse a realidades parecidas.

Según el tipo de datos empleados, se aplicará la Investigación Cuantitativa ya que analiza los fenómenos a través de la recolección de datos y se apoya en la utilización de herramientas estadísticas, matemáticas e informáticas para calibrarlos.

Según el grado de manipulación de variables, es una investigación No Experimental, con Diseño Transversal o Transeccional, Correlacional-Causal, porque no se manipulará deliberadamente variables. Dicho de otra forma, son estudios en los que no hacemos variar en forma intencional las variables independientes para ver su efecto sobre otras variables, asimismo los datos se recolectarán de manera inmediata, en un periodo de tiempo único, ya que su finalidad es describir variables y analizar su incidencia e interrelación en un momento específico. Consideramos que esta es una investigación Correlacional-Causal porque puede limitarse a establecer relaciones entre variables sin especificar sentido de causalidad o intentar analizar relaciones causales.

Según el tipo de inferencia consideramos que es una Investigación Hipotética-deductiva, ya que se apoya en la observación real de la formación de hipótesis.

Según el tiempo en el que se realiza, la hemos clasificado como una Investigación Transversal, ya que se aplica para observar las alteraciones que han ocurridos en los fenómenos, individuos o grupos durante un periodo de tiempo determinado.

### **2.3.Diseño de contrastación de hipótesis/procedimiento a seguir en la investigación**

El objetivo de la investigación es Implementar un prototipo de Sistema en tiempo real de bebederos de ganado vacuno para controlar nivel de agua en la Quinta Fernández, en el diseño de contrastación de la hipótesis, se medirá el impacto de la variable independiente(X): Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno, sobre la variable dependiente(Y): Controlar Nivel de Agua en la Quinta Fernández, midiendo y comparando los resultados. El diseño de contrastación de la hipótesis se formula a continuación:

$$Ge: X \Rightarrow Y$$

Dónde:

*Ge*: Grupo de Estudio

*X*: Representa el Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacunos.

*Y*: Nivel de Agua en la Quinta Fernández.

### **2.4.Población, muestra**

#### **2.4.1. Población**

La población son los trabajadores encargados del cuidado del ganado, de la Quinta Fernández que tienen problemas con el llenado de bebederos de su ganado vacuno.

#### **2.4.2. Muestra**

La muestra que usaremos es una Muestra Poblacional, se tomará la misma población de estudio, en la Quinta Fernández, ubicada en el Sector Rural “La Clake” del distrito de Reque, provincia de Chiclayo, departamento de Lambayeque.

## 2.5. Métodos Aplicados en la Investigación

Los métodos de investigación considerados en nuestro proyecto, según la clasificación de Rodríguez Jiménez & Pérez Jacinto (2017) son los siguientes:

### 2.5.1. *Métodos para la búsqueda de información*

**Histórico – Lógico.** Consideramos utilizar este método porque se aplica en la búsqueda de información con el fin de procesar lógicamente los antecedentes relacionados con el objeto de investigación; además este método se emplea a menudo cuando se buscan los antecedentes de un problema científico y mientras se elaboran las bases teóricas y metodológicas de la propuesta de solución al problema. En cualquiera de los dos casos, su propósito es encontrar la información dentro del periodo de indagación de la red.

**Sistematización.** Este método ha estado enlazado al avance del método científico. La sistematización de acuerdo con el método racional de investigación se pueden considerar dos propósitos distintos:

- i) La sistematización como herramienta de procesamiento de información, también llamado sistematización de información o datos, se indica el orden y la lista de datos e información.
- ii) la sistematización de experiencias, se refiere a los sucesos que se ejecutan en un determinado tiempo.

Por tanto, lo apropiado es hacer uso del método de sistematización de experiencias, ya que nuestro proyecto se desarrolla en un periodo determinado de tiempo.

### 2.5.2. *Métodos para construcción del conocimiento*

**La modelación.** Hemos considerado el uso de este método porque se originan modelos con el fin de investigar la realidad, teniendo como propósito construir conocimientos como parte de la configuración del sistema teórico, conceptual o procedimiento del desarrollo de la investigación.

## 2.6. Técnicas, instrumentos, equipos y materiales

Se usarán técnicas, instrumentos, equipos y materiales, así como también herramientas que permitan el diseño, construcción y la simulación del prototipo del proyecto.

### 2.6.1. Técnicas

**Observación sistemática.** Según Arias Gonzáles (2020), la observación sistemática es un procedimiento utilizado de forma cuidadosa y bajo un criterio académico para recopilar información del objeto de estudio; en otras palabras, se considera la acción de observar, interpretar, examinar y, a través de un proceso sistemático mediante el cual se obtienen las conclusiones de dicha técnica. (p.71)

**Entrevista.** Según Herrera Guaítara (2011), es una técnica aplicada a la recopilación de datos a través de un sistema de interrogantes, mediante la intercomunicación verbal entre dos o más personas. (p.54)

Pomposo Yanes (2015) nos dice que las entrevistas permiten a los entrevistadores y a los entrevistados discutir las interpretaciones y expresar sus respectivos puntos de vista.(p.68)

### 2.6.2. Instrumentos

**Guía de Observación.** Patricia Criollo (2021), nos indica que la guía de observación es un instrumento basado en una lista de indicadores que se pueden elaborarse como preguntas o afirmaciones; esta guía es un instrumento que permite al observador ubicarse sistemáticamente para ver cuál es realmente el objeto de estudio para la investigación; además es el medio que canaliza a la recopilación y obtención de datos e información sobre un fenómeno o hecho.

La Guía de Observación consiste en cuatro preguntas, dichas preguntas se harán para poder medir la calidad del software. La Guía de Observación se realizará luego de ser construido el software y se aplicará a los trabajadores de la Quinta Fernández.

**Guía de Entrevista.** Según Ortiz Cruz (2015) es un documento que abarca temas, preguntas sugerida y aspectos a estudiar en una entrevista.

La Guía de Entrevista constará de seis ítems, que se aplicará a los encargados del cuidado del ganado vacuno de la Quinta Fernández para identificar la problemática con respecto al control del nivel de agua en los bebederos.

### 2.6.3. Equipos

**Equipo Humano.** Está conformado por los profesionales que desarrollarán el presente proyecto de investigación.

**Equipo Tecnológico.** Para cumplir nuestro propósito usamos:

- Laptops
- USBs

#### **2.6.4. Materiales**

**Sensor de turbidez de agua.** Naylamp Mechatronics (n.d.-b), afirma que el sensor de turbidez de agua está formado por una alimentación de corriente de 5V DC, una corriente máxima de funcionamiento de 40mA, 4 Pines de conexión, una temperatura de funcionamiento de 5 °C a 90 °C, además presenta una salida analógicas y una digital.

**Figura 2**

*Sensor de turbidez de agua.*



**Nota.** Este sensor nos permite ver la turbidez del agua del bebedero. Tomada de (Naylamp Mechatronics, n.d.-g)

**Sensor de nivel de líquido horizontal.** Según Naylamp Mechatronics, (n.d.-f), tiene como especificaciones técnicas un voltaje máximo de 100V corriente continua/alterna, una corriente máxima de 0.5A, carga máxima de 10W, un cable de 30 centímetros y un sensor de nivel modelo boya o interruptor flotador.

**Figura 3**

*Sensor de nivel de líquido horizontal.*



**Nota.** Este sensor nos permite conocer el nivel de agua del bebedero. Tomada de (Naylamp Mechatronics, n.d.-f)

**Placa ESP32.** Naylamp Mechatronics, (n.d.-d), nos menciona que la placa ESP32, posee un voltaje de alimentación de 5V de corriente directa, y su voltaje de entrada y salida es de 3.3V, esta potente placa presenta una memoria ROM de 448 Kbyte, SRAM 520 KByte y SRAM in RTC de 16 Kbyte, además tiene un CPU Dual-Core Tensilica Xtensa LX6 de 32 bit, wifi, bluetooth y 30 pines.

**Figura 4**

*NODEMCU-32 30-PIN ESP32 WIFI*



**Nota.** La placa ESP32 es una solución de Wi-Fi/Bluetooth con un procesador integrado que puede conectarse con varios periféricos. Tomada de (Naylamp Mechatronics, n.d.-f)

**Válvula solenoide ½" 12 VDC.** Naylamp Mechatronics (n.d.-a) nos afirma que este dispositivo es adecuado para agua y fluidos de baja densidad, presenta voltaje de operación de 12V de corriente continua, potencia de consumo de 8W, un conector de tubería con rosca externa ½" NPS Macho, el material del cuerpo es plástico ABS con dimensiones de 85x60x26 mm.

**Figura 5**

*Válvula solenoide ½" 12 VDC.*



**Nota.** Sensor para controlar el flujo del agua. Tomada de (Naylamp Mechatronics, n.d.-j)

**Fuente de alimentación DC 12V 1A.** Según Naylamp Mechatronics (n.d.-b) esta fuente posee un voltaje de entrada de 100V a 240V de corriente alterna y un voltaje de salida de 12V de corriente continua, corriente de salida máxima de 1A, potencia máxima de 12W, con eficiencia de salida al 80%, un led indicador de encendido, un enchufe de corriente alterna tipo A, conector de salida de corriente continua, con longitud de 90 centímetros de cable aproximadamente, y presenta unas dimensiones de 75\*65\*25mm aproximadamente (L\*WH), tiene un peso de 63 gramos, una temperatura de operación de -10° hasta +60°C y un modelo de XRD-1210.



**Figura 6***Fuente de alimentación DC 12V 1A*

**Nota.** Esta fuente suministra electricidad a los dispositivos. Tomada de (Naylamp Mechatronics, n.d.-b)

**Fuente de alimentación DC 5V 1A.** Naylamp Mechatronic (n.d.) nos indica que tiene un voltaje de entrada de 100V a 240V de corriente alterna (50/60Hz), voltaje de salida de 5V de corriente continua, corriente de salida máxima de 1A, potencia máxima de 5W, eficiencia de salida de 80%, un led indicador de encendido, enchufe de corriente alterna tipo A, conector de salida de corriente continua, con longitud de 90 centímetros de cable (aproximadamente), con dimensiones: 58\*65\*20mm aproximadamente (L\*WH), tiene un peso de 50 gramos, con temperatura de operación: -10° hasta +60°C, cuyo modelo es 0510.

**Figura 7***Fuente de alimentación DC 5V 1A*

**Nota.** Esta fuente suministra electricidad a los dispositivos. Tomada de (Naylamp Mechatronic, n.d.)

**Módulo Relé 1CH 5VDC.** Naylamp Mechatronics (n.d.-c) nos indica que tiene una capacidad de operación de 5V de corriente continua, capacidad máxima de 10A (NO), 5A (NC), tiempo de acción de 10ms / 5ms y para activar salida NO de 0 Voltios.

**Figura 8**

*Módulo Relé 1CH 5VDC*



**Nota.** Funciona como interruptor de equipos de alta potencia. Tomada de (Naylamp Mechatronics, n.d.-c)

**Sensor Ultrasónico.** Según lo indicado por Naylamp Mechatronics, (n.d.-i), este pequeño sensor mide 45\*20\*15 mm, presenta una frecuencia acústica de 40KHz, un voltaje de alimentación de 5V y una corriente de alimentación 15mA.

**Figura 9**

*Sensor Ultrasónico.*



**Nota.** Detecta y mide la distancia. Tomada de (Naylamp Mechatronics, n.d.-h)

**Interruptor de botón de enclavamiento.** Naylamp Mechatronics (n.d.-e) nos dice que es un pulsador tipo enclavamiento, de color rojo, de plástico.

**Figura 10**

*Interruptor de botón de enclavamiento*



**Nota.** Interruptor tipo enclavamiento. Tomada de (Naylamp Mechatronics, n.d.-d)

### Capítulo III. Desarrollo de la Propuesta.

#### 3.1.Comunicación.

##### 3.1.1. Aplicación de instrumentos para la recolección de datos

Para aplicar el instrumento de recolección de datos, se realizó una reunión con el señor Simeón Tarrillo Díaz, encargado de controlar el llenado de bebederos en la quinta Fernández, a quien se le aplicó una entrevista, con seis ítems para recolectar los datos y poder definir los requerimientos del software.

En el siguiente enlace podrán visualizar la entrevista realizada al señor Simeón Tarrillo Guevara en la Quinta Fernández del distrito de Reque. [https://drive.google.com/drive/folders/1REsKWuYqDq\\_QfnITp3PF97wZ\\_mU6uaZR?usp=sharing](https://drive.google.com/drive/folders/1REsKWuYqDq_QfnITp3PF97wZ_mU6uaZR?usp=sharing).

##### 3.1.2. Análisis de requisitos

Identificación de los requisitos funcionales y no funcionales del sistema.

#### Requisitos Funcionales.

Son los comportamientos del software, las funcionalidades esperadas para el sistema informático, es decir lo que el sistema debe hacer.

**Tabla 1**

*Requisitos Funcionales del Sistema*

Código	Requisito Funcional
RF1	Llenar bebedero
RF2	Detener llenado cuando el agua esté sucia
RF3	Verificar nivel de agua actual
RF4	Detener llenado por nivel máximo

**Nota.** Esta tabla muestra los requisitos funcionales del sistema en tiempo real.

**Caso de Uso Llenar bebedero.** Este caso de uso comienza cuando el bebedero se encuentra en su nivel mínimo, y necesita ser llenado.

**Tabla 2**

*Caso de Uso Llenar Bebedero.*

<b>CUS01: LLENAR BEBEDERO</b>		
<b>DESCRIPCIÓN:</b> El caso de uso comienza cuando el bebedero necesita ser llenado.		
<b>ACTOR:</b>	Ganadero	
<b>PRECONDICIÓN:</b>	Que el bebedero esté en su nivel mínimo de agua	
<b>SECUENCIA:</b>	<b>PASO</b>	<b>EVENTO</b>
<b>FLUJO PRINCIPAL</b>	1	El sistema muestra al ganadero que el nivel de agua se encuentra en el nivel mínimo
	2	Luego el bebedero se llenará de manera automática hasta el nivel máximo.
<b>FLUJO ALTERNATIVO</b>	1.1	Otra alternativa es que el ganadero ingrese a la aplicación Blynk donde dará clic en el botón para llenar el bebedero.
<b>POSTCONDICIONES</b>	Ninguno	

**Nota.** Esta tabla muestra el caso de uso Llenar Bebedero.

**Caso de Uso Detener llenado cuando el agua este sucia.** El caso de uso comienza cuando el sensor de turbidez detecta que el agua está sucia.

**Tabla 3**

*Caso de Uso Detener Llenado Cuando el Agua esté Sucia*

<b>CUS02: DETENER LLENADO CUANDO EL AGUA ESTÉ SUCIA</b>		
<b>DESCRIPCIÓN:</b> El caso de uso comienza cuando el sensor de turbidez detecta que el agua está sucia.		
<b>ACTOR:</b>	Ganadero	
<b>PRECONDICIÓN:</b>	Que el sensor detecte que el agua está sucia	
<b>SECUENCIA:</b>	<b>PASO</b>	<b>EVENTO</b>
<b>FLUJO PRINCIPAL</b>	1	El sistema enviará un correo al ganadero que el agua está sucia y necesita ser cambiada.
	2	El sistema cerrará el flujo del agua.
	3	El ganadero le da clic en la opción desaguar de la aplicación Blynk.
	4	El sistema comienza a vaciar el bebedero.
<b>FLUJO ALTERNATIVO</b>	-	Ninguno
<b>POSTCONDICIONES</b>	Llenar bebedero con el visto bueno del ganadero.	

**Nota.** Esta tabla muestra el caso de uso Detener llenado cuando el Agua esté Sucia.

*Caso de Uso Verificar nivel de agua actual.* El caso de uso comienza cuando el ganadero ingresa a la aplicación Blynk y visualiza el nivel de agua.

**Tabla 4***Caso de Uso Verificar Nivel de Agua Actual*

<b>CUS03: VERIFICAR NIVEL DE AGUA ACTUAL</b>		
<b>DESCRIPCIÓN:</b> El caso de uso comienza cuando el ganadero ingresa a la aplicación Blynk y visualiza el nivel del agua.		
<b>ACTOR:</b>	Ganadero	
<b>PRECONDICIÓN:</b>	Ninguno	
<b>SECUENCIA:</b>	<b>PASO</b>	<b>EVENTO</b>
<b>FLUJO PRINCIPAL</b>	1	El ganadero ingresa a la aplicación Blynk
	2	El sistema visualiza el nivel de agua que contiene el bebedero.
	-	Ninguno
<b>FLUJO ALTERNATIVO</b>	Ninguno	
<b>POSTCONDICIONES</b>		

**Nota.** Esta tabla muestra el caso de uso Verificar nivel de agua actual.

*Caso de Uso Detener llenado por nivel máximo.* El caso de uso comienza cuando el ganadero desea detener el llenado del bebedero cuando ya está en su nivel máximo.

**Tabla 5**

*Caso de Uso Detener Llenado por nivel máximo.*

<b>CUS04: DETENER LLENADO POR NIVEL MÁXIMO</b>		
<b>DESCRIPCIÓN:</b> El caso de uso comienza cuando el ganadero desea detener el llenado del bebedero.		
<b>ACTOR:</b>	Ganadero	
<b>PRECONDICIÓN:</b>	Que el bebedero alcance el nivel máximo de llenado.	
<b>SECUENCIA:</b>	<b>PASO</b>	<b>EVENTO</b>
<b>FLUJO PRINCIPAL</b>	1	El sistema detecta que el nivel del agua se encuentra al máximo.
	2	El sistema detendrá el llenado automáticamente.
<b>FLUJO ALTERNATIVO</b>	Ninguna	Otra alternativa es que el ganadero ingrese a la aplicación Blynk donde dará clic en el botón para detener el bebedero.
<b>POSTCONDICIONES</b>		

**Nota.** Esta tabla muestra el caso de uso Detener llenado por nivel máximo.

#### **Requisitos No Funcionales.**

Se enfocan en el diseño y la implementación, en las propiedades del sistema, es decir hacen referencia a las cualidades, restricciones y características del software.



**Tabla 6***Requisitos No Funcionales del Sistema*

<b>Código</b>	<b>Requisito No Funcional</b>
RNF1	El sistema será codificado con el lenguaje de programación Arduino en el entorno de desarrollo Arduino
RNF2	Se usará la App Blynk (plataforma de IoT) para controlar y supervisar el sistema de manera remota.
RNF3	Es un sistema de entrada y salida

**Nota.** Esta tabla muestra los requisitos no funcionales del sistema en desarrollo.

### 3.2. Plan Rápido

#### 3.2.1. Análisis de costos y presupuestos

**Tabla 7**

*Materiales usados en el proyecto*

<b>Cantidad</b>	<b>Descripción</b>	<b>Precio Unitario (S/)</b>	<b>Total (S/)</b>
<b>1</b>	Sensor de Turbidez de Agua	110.00	110.00
<b>2</b>	Sensor de nivel de líquido horizontal o Boya	10.00	20.00
<b>1</b>	Placa ESP32	60.00	60.00
<b>1</b>	Válvula solenoide ½" 12 VDC	25.00	25.00
<b>1</b>	Válvula solenoide ¾" 220VAC baja presión	120.00	120.00
<b>1</b>	Fuente de alimentación DC 12V 1 <sup>a</sup>	10.00	10.00
<b>1</b>	Fuente de alimentación DC 5V 1 <sup>a</sup>	10.00	10.00
<b>2</b>	Módulo Relé 1CH 5VDC	6.00	12.00
<b>1</b>	Interruptor de botón de enclavamiento	2.00	4.00
<b>1</b>	Sensor Ultrasónico	8.00	8.00
<b>3</b>	Paquete cable dupont 20cm HH/HM/MM	5.00	15.00
<b>1</b>	Cable de cobre de colores 6 m	5.00	5.00
<b>1</b>	Impresión de placa	50.00	50.00
<b>1</b>	Borneras, header hembra y machos	4.40	4.40
<b>TOTAL (S/)</b>			<b>453.40</b>

**Nota.** Esta tabla muestra los materiales que serán necesario para la elaboración del prototipo.

**Tabla 8***Costos del hardware.*

Descripción	Tiempo	Precio (S/)	Total (S/)
Laptops	11 meses	S/. 1,200.00	S/. 1,200.00
<b>TOTAL (S/)</b>			<b>S/.1,200.00</b>

**Nota.** Esta tabla muestra los costos del hardware

**Tabla 9***Servicios.*

Descripción	Tiempo	Precio (S/)	Total (S/)
Internet	11 meses	S/. 50.00	S/. 550.00
Telefonía	11 meses	S/. 30.00	S/. 330.00
Otros	11 meses	S/. 20.00	S/.220.00
<b>TOTAL (S/)</b>			<b>S/.1100.00</b>

**Nota.** Esta tabla muestra los gastos en servicios

**Tabla 10***Costos totales.*

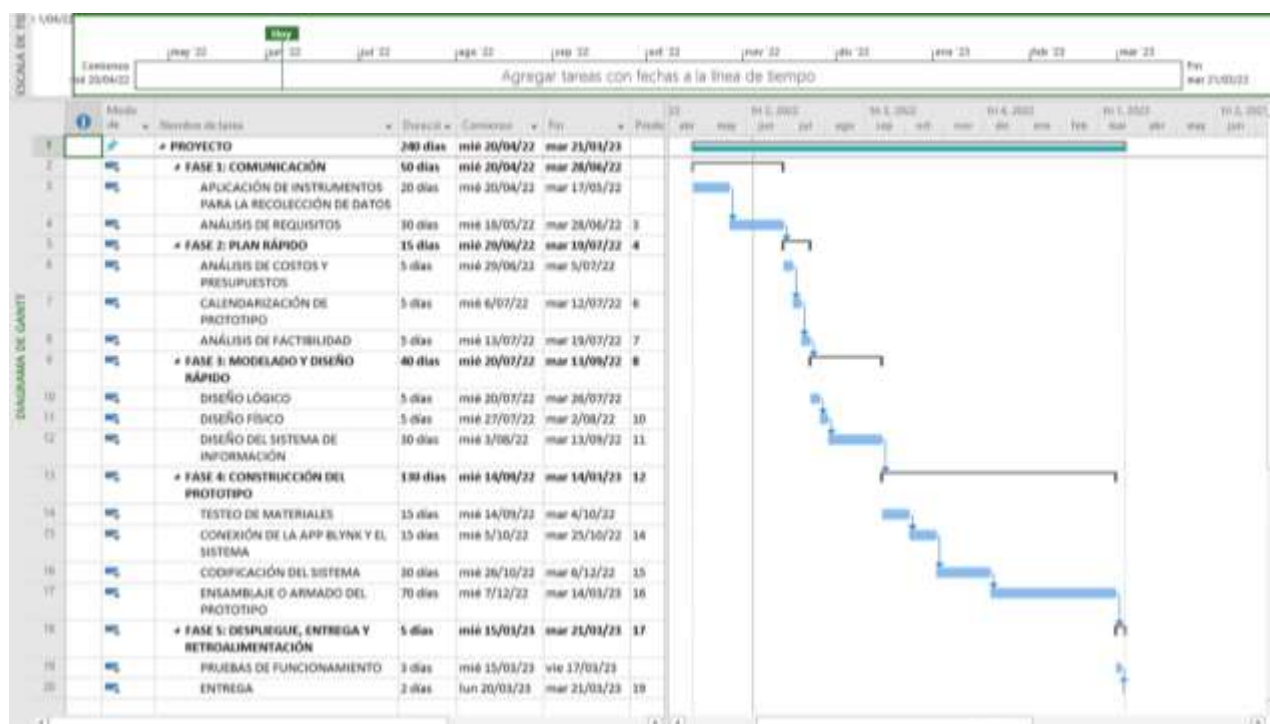
Descripción	Totales (S/)
Materiales	S/.453.40
Costos de Hardware	S/. 1,200.00
Servicios	S/. 1100.00
<b>TOTAL</b>	<b>S/. 2753.40</b>

**Nota.** Esta tabla muestra los costos totales de la elaboración del proyecto

### 3.2.2. Calendarización del prototipo

**Figura 11**

*Calendarización del prototipo*



**Nota.** Cronograma de fases del proyecto. Elaboración propia.

### 3.2.3. *Análisis de factibilidad*

#### **Justificación técnica y económica**

**Tabla 11**

*Justificación técnica y económica*

<b>Requisitos</b>	<b>Justificación Técnica</b>	<b>Justificación Económica</b>
Llenar bebedero	Se usará una válvula solenoide, que permitirá controlar el flujo del agua y así tener el control del llenado del bebedero.	Se comprará una válvula solenoide ½” 12 V DC
Detener llenado cuando el agua esté sucia	Se hará uso de un sensor de turbidez de agua, que permitirá verificar la pureza del agua y dependiendo de eso se detendrá el llenado.	Se comprará un sensor de turbidez de agua.
Consultar nivel de agua actual	Se colocará un Sensor Ultrasónico a prueba de agua que nos permitirá saber a qué nivel de agua se encuentra el bebedero, y a través de la app se podrá interactuar con el ganadero, el cual podrá realizar la consulta	Se comprará un Sensor Ultrasónico y se hará uso de la aplicación Blynk de uso gratuito
Detener llenado por nivel máximo	Se hará uso de un sensor de líquido horizontal o boya, para determinar si el bebedero se encuentra en su máximo nivel de llenado.	Se comprará un sensor de líquido horizontal o boya.
Monitorear de manera remota	Se vinculará la App Blynk (plataforma IoT) con el sistema en tiempo real.	El costo será nulo ya que usaremos su versión gratuita.

**Nota.** Esta tabla muestra la justificación técnica y económica de la propuesta.

### Justificación tecnológica

En nuestro proyecto se puede utilizar la placa Arduino 33 IoT o la ESP32, de las cuales hemos creído conveniente hacer uso de la placa ESP32 por que ambas tienen funciones similares, evaluamos los costos de ambas y se nos hizo más factible comprar la ESP32, ya que en nuestro país no está disponible la placa Arduino 33 IoT, y se tendría que hacer pedido al extranjero, y como ustedes saben la situación de ahora está un poco complicada, es el motivo por el cual optamos por quedarnos con la placa ESP32 que si se puede encontrar en nuestro país.

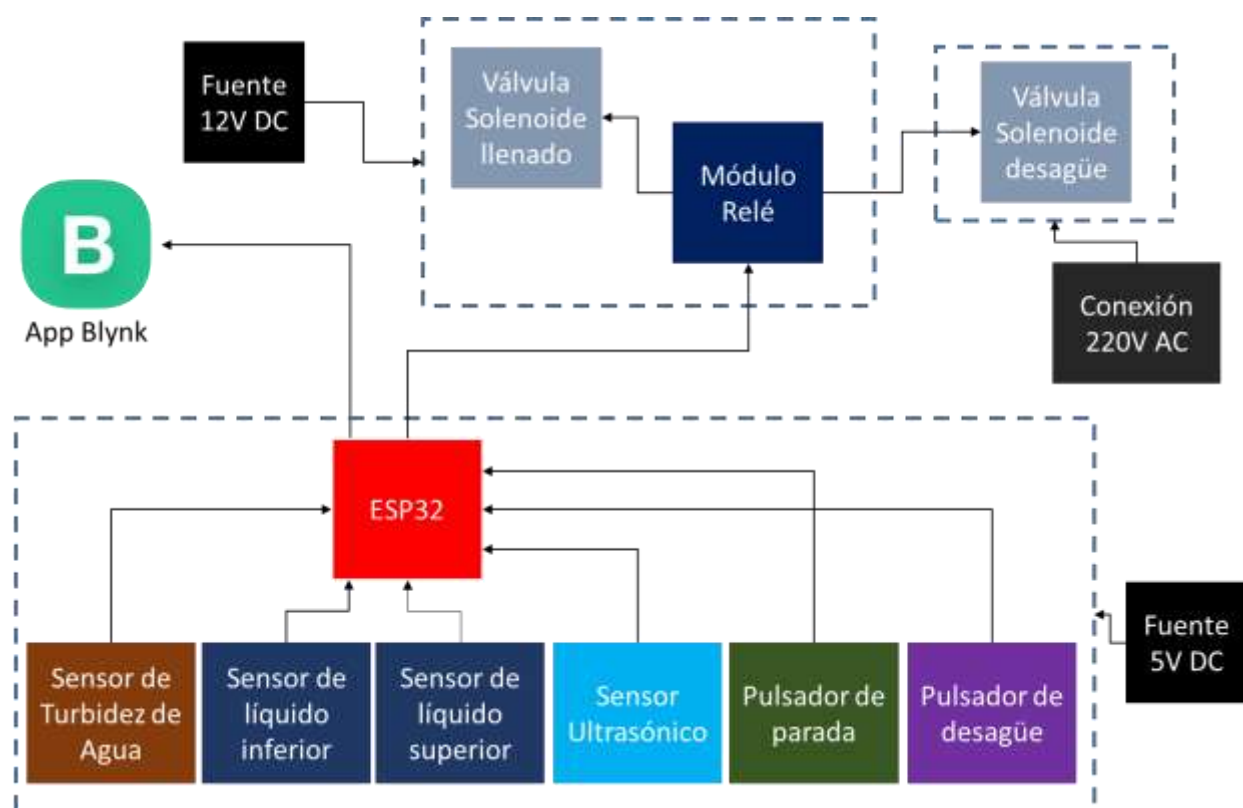
### 3.3. Modelado y diseño rápido

#### 3.3.1. Diseño lógico

A continuación, se muestra el diseño lógico a través de un diagrama de bloques empleado para explicar la interacción entre ellos y el funcionamiento interno del sistema en tiempo real.

**Figura 12**

*Diagrama de Bloques*



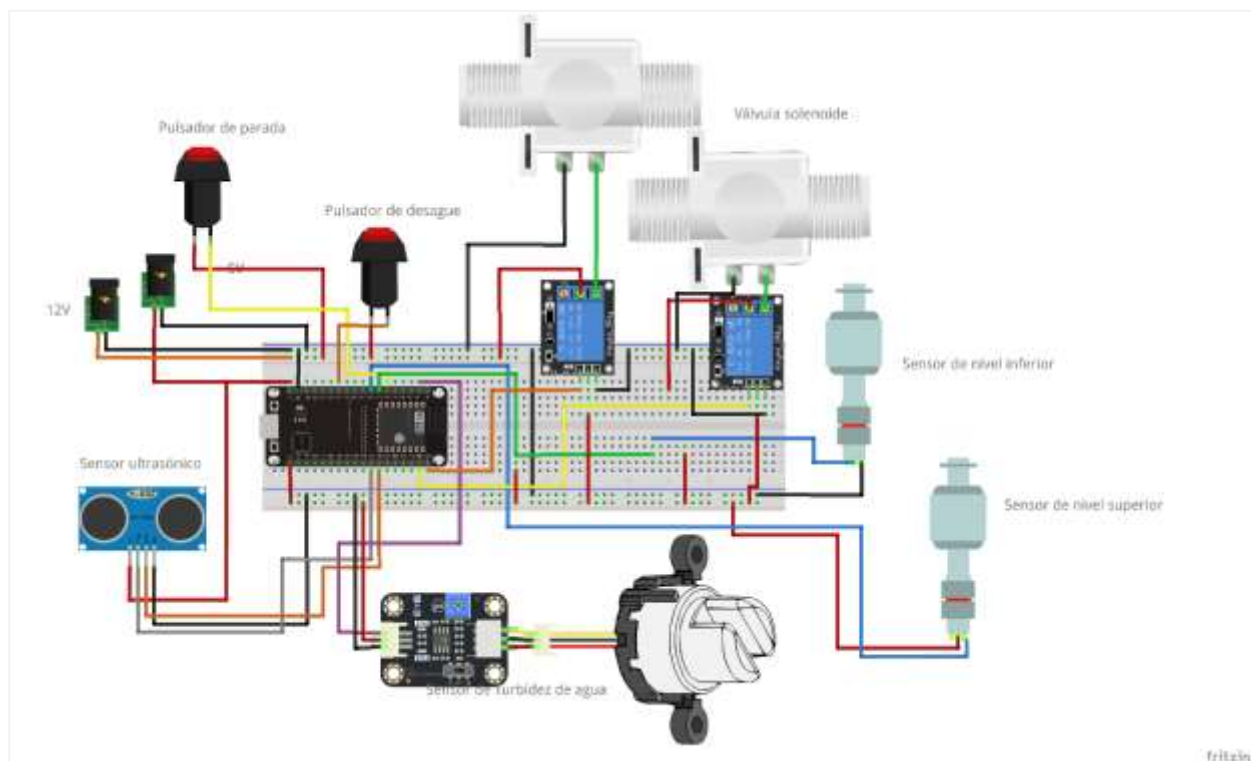
**Nota.** Representación del funcionamiento interno del proyecto. Elaboración propia.

### 3.3.2. *Diseño físico*

Para realizar el diseño físico hemos hecho uso del programa de automatización de diseño electrónico Fritzing, donde se muestran los componentes empleados y las conexiones que se deben realizar en el prototipado.

**Figura 13**

*Diagrama de Componentes*



**Nota.** El diagrama de componentes proporciona una visión orientada a la ejecución del proyecto. Elaboración propia.

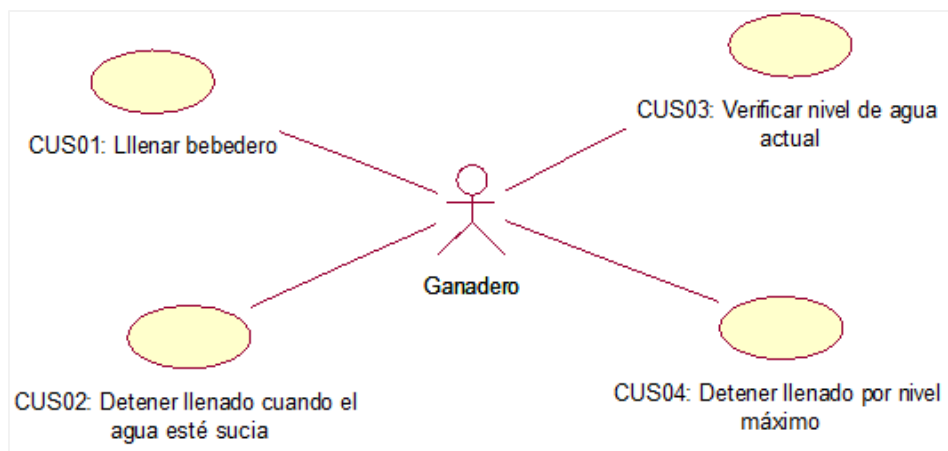
### 3.3.3. *Diseño del sistema de información (diagramas de casos de uso, secuencia o actividad)*

#### **Modelo de Casos de Uso del Sistema**

En la siguiente figura se muestra el diagrama de casos de uso resultante de la especificación de requisitos funcionales.

**Figura 14**

*Modelo de Casos de Uso del Sistema (modelo o modelado)*



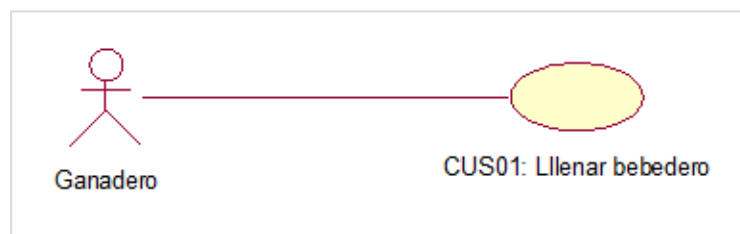
**Nota.** Los requisitos funcionales que el ganadero va a realizar. Elaboración propia.

### **Diagramas de Casos de Uso del Sistema.**

- ***CUS01: Llenar bebedero.***

**Figura 15**

*CUS01: Llenar bebedero*

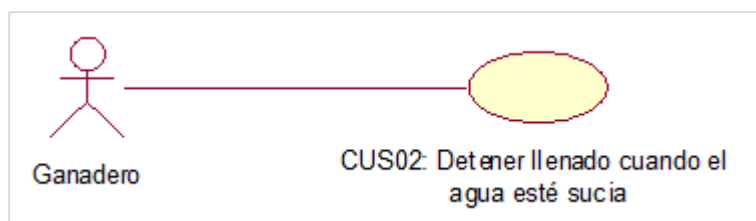


**Nota.** Primer caso de uso, Llenar bebedero. Elaboración propia.

- ***CUS02: Detener llenado cuando el agua esté sucia.***

**Figura 16**

*CUS02: Detener llenado cuando el agua esté sucia.*



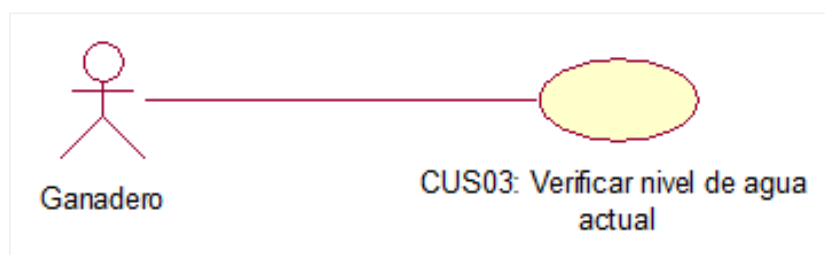
**Nota.** Segundo caso de uso, detener llenado cuando el agua esté sucia. Elaboración propia.



- **CUS03: Verificar nivel de agua actual.**

**Figura 17**

*CUS03: Verificar nivel de agua actual.*

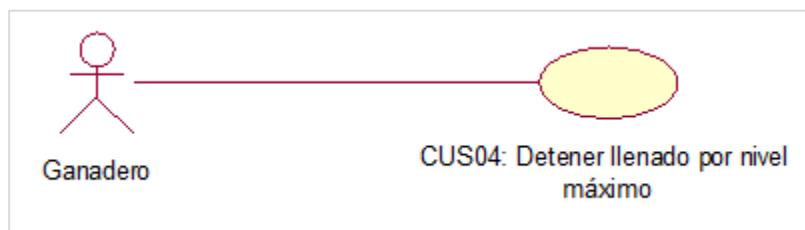


**Nota.** Tercer caso de uso, verificar nivel de agua actual. Elaboración propia.

- **CUS04: Detener llenado por nivel máximo.**

**Figura 18**

*CUS04: Detener llenado por nivel máximo.*



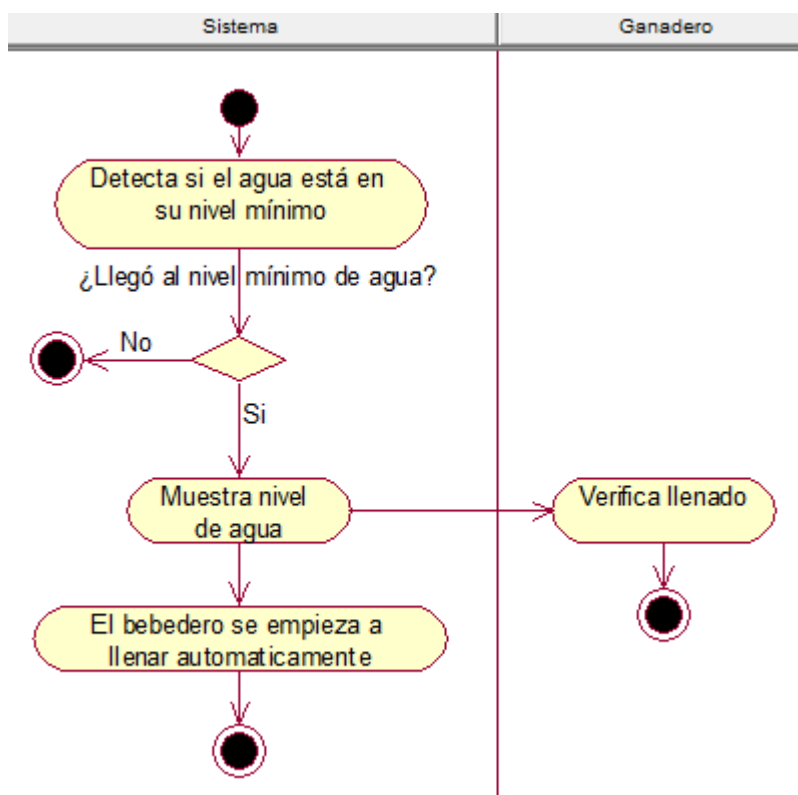
**Nota.** Cuarto caso de uso, detener llenado por nivel máximo. Elaboración propia.

### Diagramas de Actividad del Sistema.

- *Diagrama de Actividades 01: Llenar bebedero*

**Figura 19**

*Diagrama de Actividades 01: Llenar bebedero*

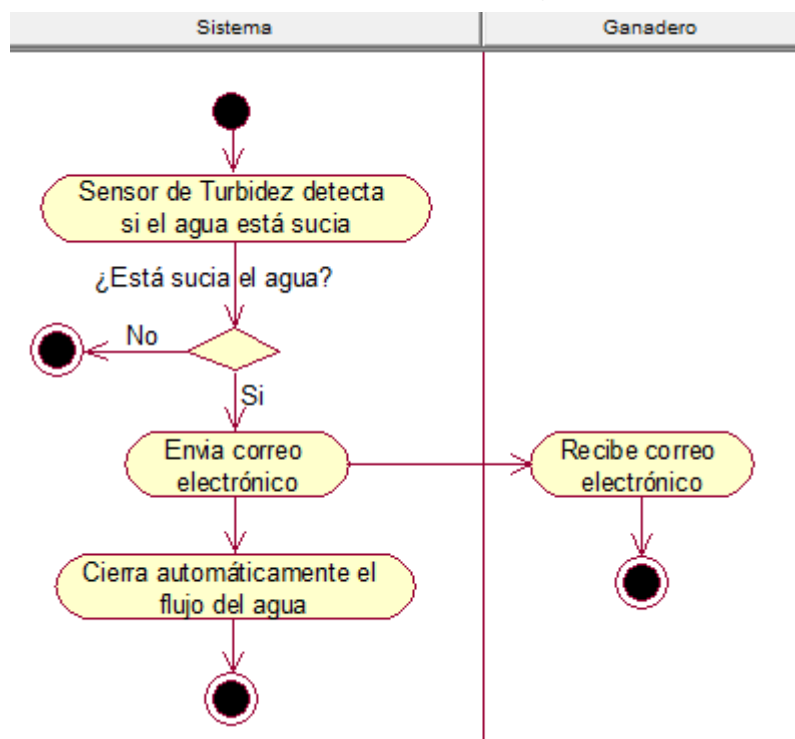


**Nota.** Diagrama de Actividades 01: Llenar bebedero. Elaboración propia.

- **Diagrama de Actividades 02: Detener llenado cuando el agua esté sucia**

**Figura 20**

*Diagrama de Actividades 02: Detener llenado cuando el agua esté sucia*

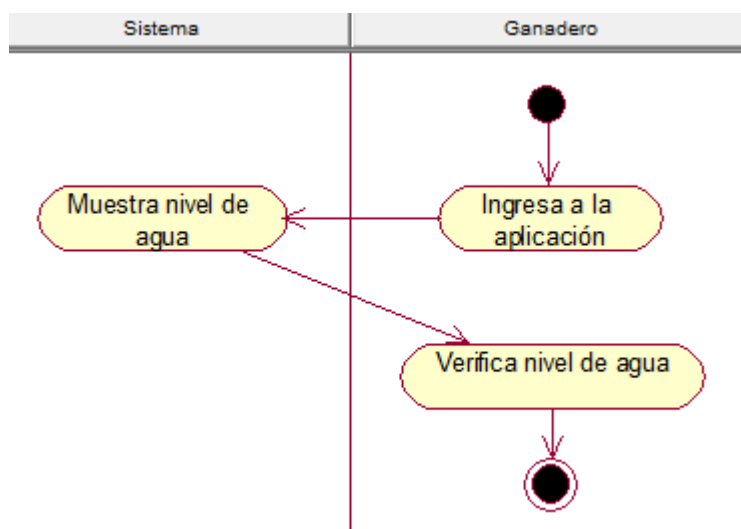


**Nota.** Diagrama de Actividades 02: Detener llenado cuando el agua esté sucia. Elaboración propia.

- *Diagrama de Actividades 03: Verificar nivel de agua actual*

**Figura 21**

*Diagrama de Actividades 03: Verificar nivel de agua actual*

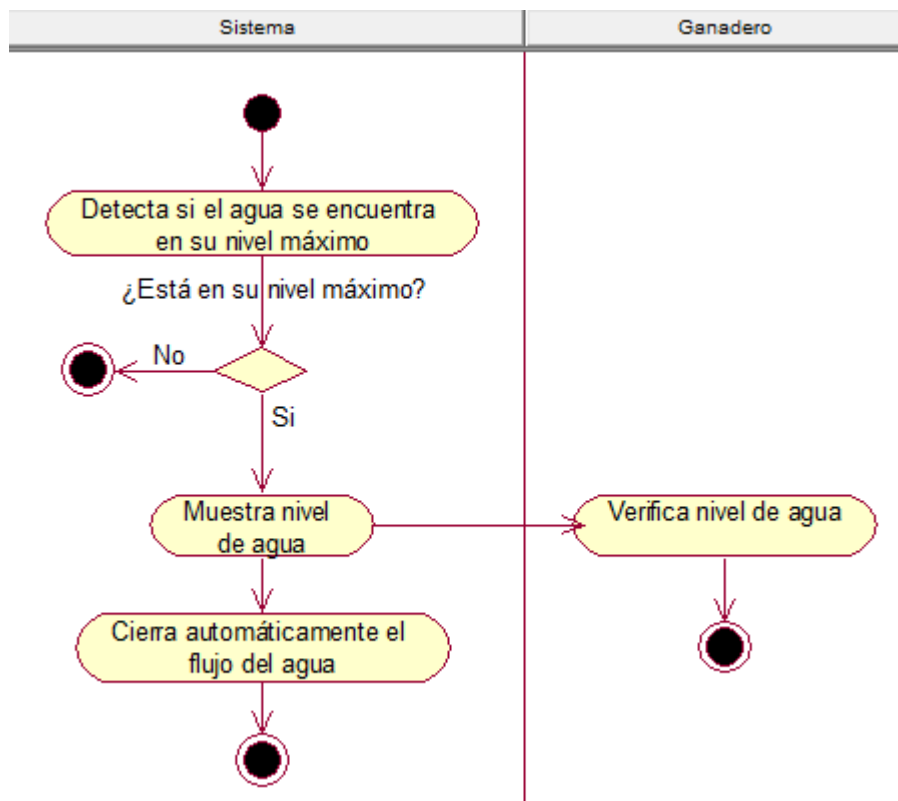


**Nota.** Diagrama de Actividades 03: Verificar nivel de agua actual. Elaboración propia.

- **Diagrama de Actividades 04: Detener llenado por nivel máximo**

**Figura 22**

*Diagrama de Actividades 04: Detener llenado por nivel máximo.*



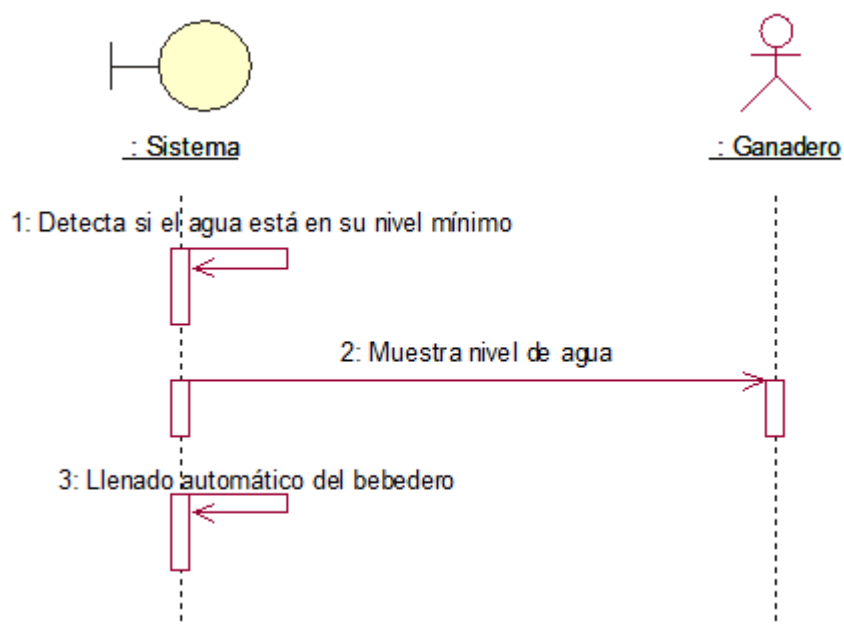
**Nota.** Diagrama de Actividades 04: Detener llenado por nivel máximo. Elaboración propia.

### Diagrama de Secuencia.

- *Diagrama de Secuencia 01: Llenar bebedero*

**Figura 23**

*Diagrama de Secuencia 01: Llenar bebedero*

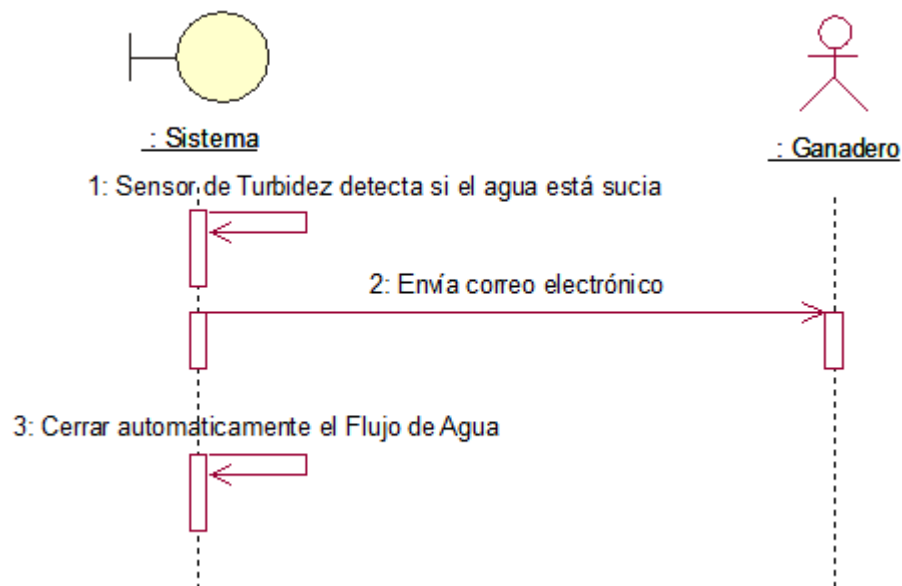


**Nota.** Diagrama de Secuencia 01: Llenar bebedero. Elaboración propia.

- **Diagrama de Secuencia 02: Detener llenado cuando el agua esté sucia**

**Figura 24**

*Diagrama de Secuencia 02: Detener llenado cuando el agua esté sucia*

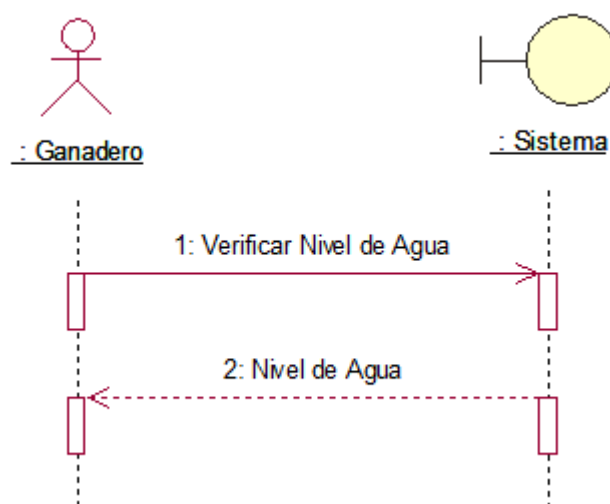


**Nota.** Diagrama de Secuencia 02: Detener llenado cuando el agua esté sucia. Elaboración propia.

- **Diagrama de Secuencia 03: Verificar nivel de agua actual**

**Figura 25**

*Diagrama de Secuencia 03: Verificar nivel de agua actual*

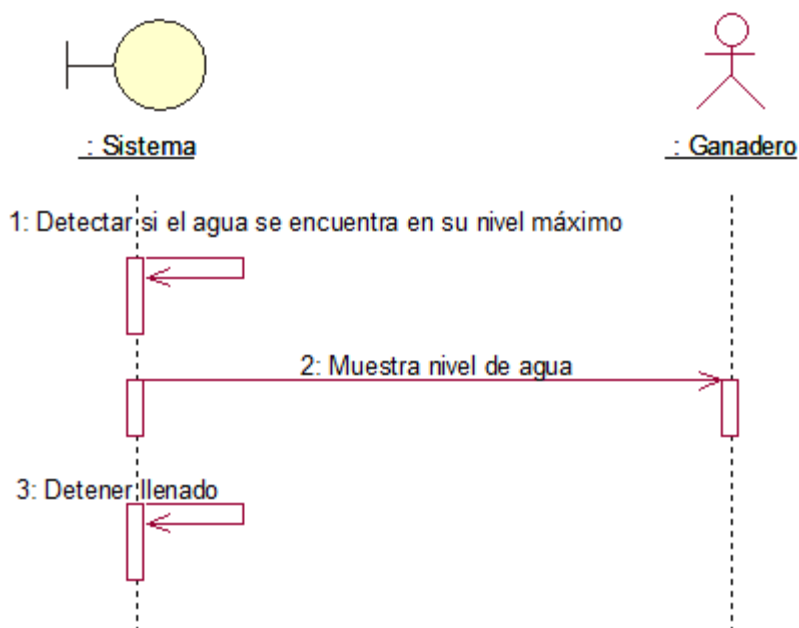


**Nota.** Diagrama de Secuencia 03: Verificar nivel de agua actual. Elaboración propia.

- **Diagrama de Secuencia 04: Detener llenado por nivel máximo**

**Figura 26**

*Diagrama de Secuencia 04: Detener llenado por nivel máximo*



**Nota.** Diagrama de Secuencia 04: Detener llenado por nivel máximo. Elaboración propia.

### 3.4.Construcción del prototipo

#### 3.4.1. Testeo de materiales

##### 3.4.1.1. Válvula Solenoide y Relé

La válvula Solenoide se usa acompañado del Módulo Relé; la válvula trabaja a 12V DC y la placa ESP32 soporta niveles de voltaje de 3.3 V DC. Es por eso que es necesario su uso, ya que nos ayuda a controlar el circuito que tiene un consumo considerable de electricidad mediante un circuito de pequeña potencia.

Como se aprecia en la imagen, hemos definido la constante reléPin, la cual hace referencia al pin 23 del ESP32 con el que se controlará el Relé, inicializándola como OUTPUT (salida digital), luego en el bucle loop enviamos una señal digital de 1 (HIGH), mediante el pin 23, la válvula se abre, y al enviar un 0 (LOW), la válvula se cierra, en un intervalo de tiempo de 1000ms (1 segundo).



**Figura 27**

*Testeo del control del módulo Relé mediante señal digital en la placa ESP32*



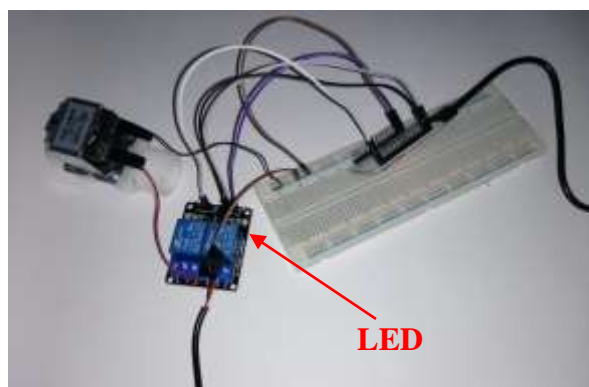
**Nota.** Codificación del testeo del control del módulo Relé mediante señal digital en la placa ESP32. Elaboración propia.

Luego de codificar y testear, llegamos a la conclusión de la Válvula Solenoide, el Módulo Relé y la placa ESP32, funcionan de manera adecuada.

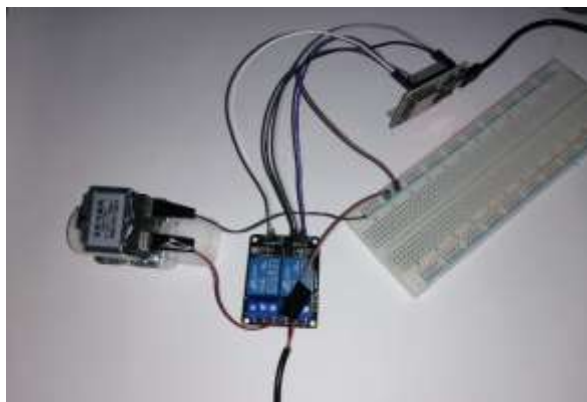
El Relé cuando está activado se enciende el led entonces la válvula se abre, cuando está desactivado pasa todo lo contrario, el led se apaga y la válvula se cierra.

**Figura 28**

*Relé Activado*



**Nota.** Relé activado, la válvula se abre. Elaboración propia.

**Figura 29***Relé Desactivado*

**Nota.** Relé desactivado, la válvula se cierra. Elaboración propia.

#### 3.4.1.2. Sensor de Nivel de Líquido Horizontal

También llamado Boya, este sensor nos permite conocer el mínimo y máximo nivel de agua que se encuentra en el bebedero, para ello hemos realizado la siguiente codificación:

**Figura 30***Test del control del sensor de líquido horizontal con la placa ESP32*

```

esp32_sensor_nivelino Arduino 1.8.9
Archivo Editar Programa Herramientas Ayuda

esp32_sensor_nivelino
#define sensorPin 33

void setup() {
  Serial.begin(9600);
  pinMode(sensorPin, INPUT);
}

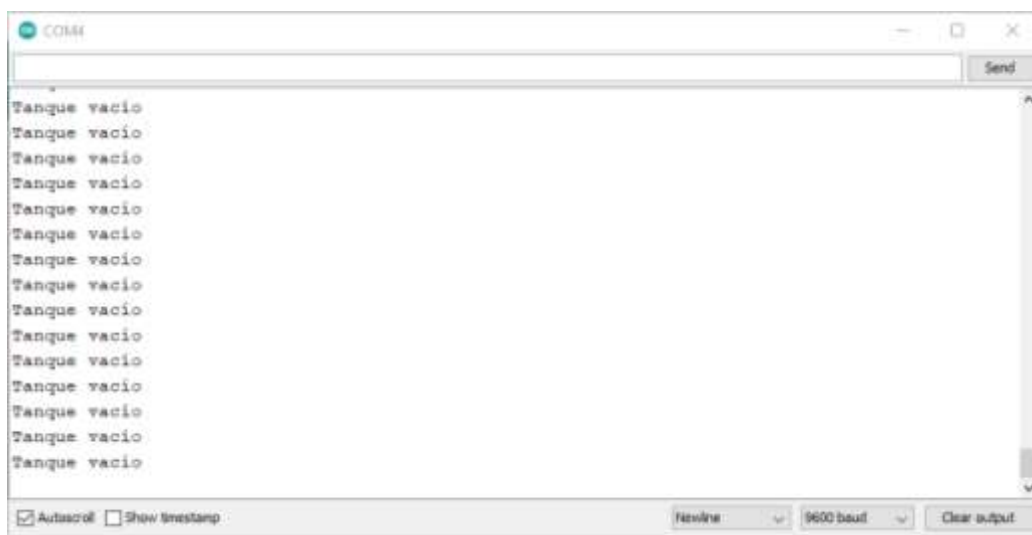
void loop() {
  if(digitalRead(sensorPin)){
    Serial.println("Tanque lleno");
  }else{
    Serial.println("Tanque vacio");
  }
  delay(1000);
}
  
```

**Nota.** Codificación del test del control del sensor de líquido horizontal con la placa ESP32. Elaboración propia.

Donde se ha definido la variable `sensorPin`, la cual hace referencia al pin 33 del ESP 32, con el que se controlará en Sensor de líquido horizontal. Para ello, hemos elegido el serial 9600 por conveniencia, el cual nos permite establecer la velocidad de datos en bits por segundo (baudios) para la transmisión de datos en serie; y nos permita enviar mensajes desde nuestro ordenador a nuestra placa ESP32 (a través de USB) y también recibir mensajes desde el IDE Arduino, luego hemos inicializado la variable `sensorPin`, la cual hace referencia al pin del ESP32, como entrada (INPUT), mediante el cual recibiremos el estado del sensor pudiendo ser 1 (HIGH) cuando esté activado o 0 (LOW) cuando esté desactivado. Posteriormente en el bucle Loop realizamos el siguiente condicional, si el sensor está activado, imprimimos en el Monitor Serie "Tanque lleno", en el caso esté desactivado, imprimimos en el Monitor Serie "Tanque vacío".

### Figura 31

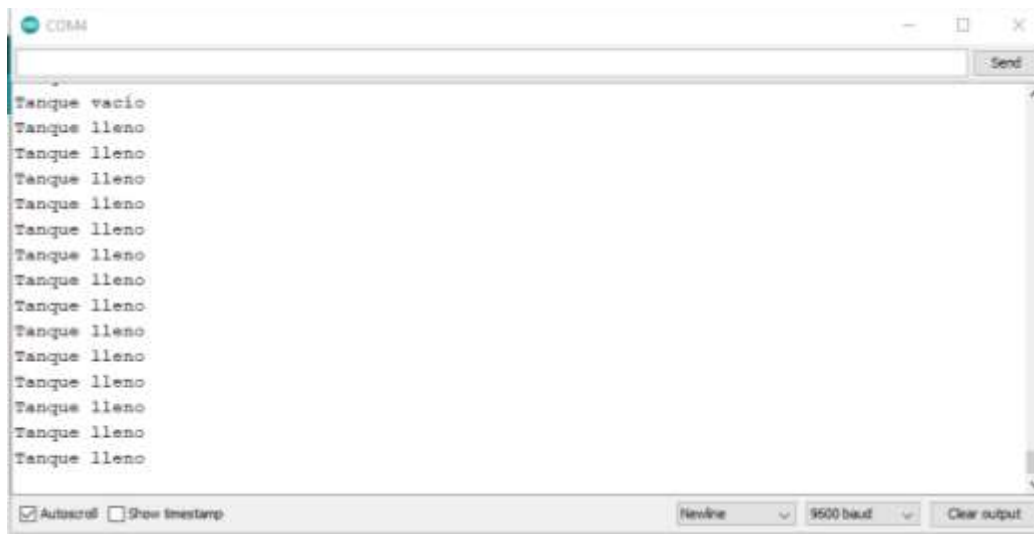
*Monitor Serie "Tanque vacío".*



**Nota.** Monitor Serie cuando el tanque está vacío. Elaboración propia.

**Figura 32**

*Monitor Serie "Tanque lleno".*



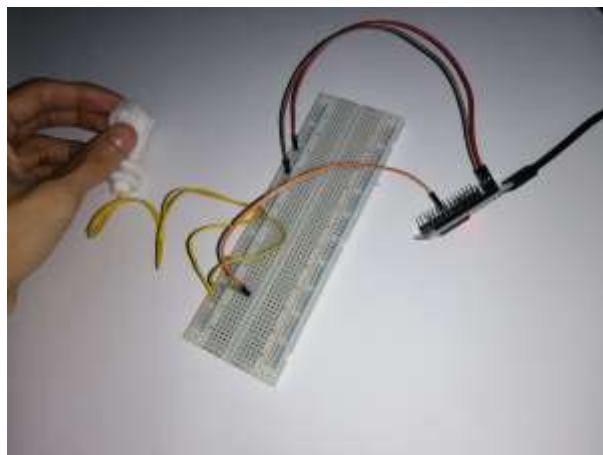
**Nota.** Monitor Serie cuando el tanque está lleno. Elaboración propia.

Al concluir la codificar y el testeo, llegamos a la conclusión el Sensor de nivel de líquido horizontal y la placa ESP32, funcionan de manera correcta.

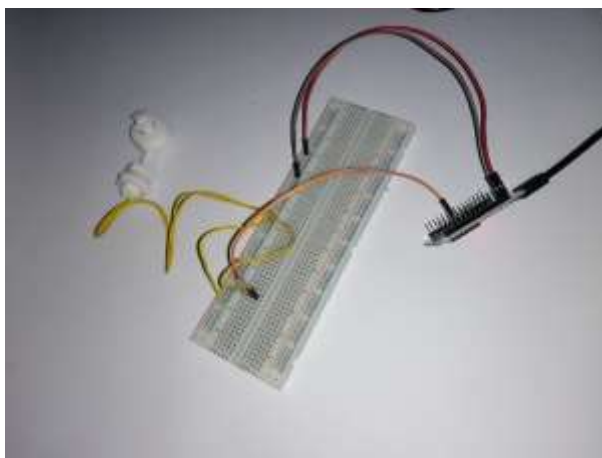
El sensor de nivel se activa cuando el aro, flota y se eleva; y se encuentra desactivado cuando está en la parte inferior.

**Figura 33**

*Sensor de nivel activado*



**Nota.** Sensor de nivel de agua activado. Elaboración propia.

**Figura 34***Sensor de nivel desactivado*

**Nota.** Sensor de nivel de agua desactivado. Elaboración propia.

#### **3.4.1.3. Sensor de Turbidez de Agua**

Como su propio nombre nos dice nos permitirá conocer la turbidez del agua del bebedero, este sensor posee un emisor y receptor infrarrojo, eso permite medir la dispersión de la luz, permitiendo detectar las partículas que se encuentran flotando en el agua, esta información será transmitido por el sensor de turbidez, mediante una variación de voltaje, la cual será medida mediante el Conversor Analógico a Digital (ADC) que posee la placa ESP32. De acuerdo a esta medición podremos saber si el agua se encuentra limpia o turbia.

Como se aprecia en la imagen hemos definido la constante `sensorPin`, la cual hace referencia al pin 36 del ESP32, después hemos elegido el serial 9600 por conveniencia, el cual nos permite establecer la velocidad de datos en bits por segundo (baudios) para la transmisión de datos en serie; y nos permita enviar mensajes desde nuestro ordenador a nuestra placa ESP32 (a través de USB) y también recibir mensajes desde el IDE Arduino.

A la constante `sensorPin` la inicializamos en INPUT (entrada). A continuación, en el bucle loop se lee la señal analógica brindada por el sensor, la remapea (reasigna) a valores entre 1 y 5, luego lo muestra en el monitor serial, repitiendo todo esto en un intervalo de un segundo.

Según nuestras mediciones, cuando el agua se encuentra limpia, el voltaje emitido por el Sensor es de 1.8V DC, por lo tanto, para calcular su valor en bits transformado por el ADC se utiliza la siguiente fórmula:

$$\text{Valor Digital} = \frac{V_{in}}{V_{ADC}} * (2^n - 1)$$

Siendo:

$V_{in}$  = Voltaje que emite el sensor de turbidez de agua

$V_{ADC}$  = Voltaje de operación del ADC

$n$  = Resolución del ADC

Donde en este caso, nuestro ADC que incorpora el ESP32 es de 12 bits de resolución, el voltaje de operación del ADC será el que brinda la placa ESP32, el cual se sitúa en 3.5V DC y el voltaje de entrada para agua limpia es 1.8V, reemplazando estos valores en la fórmula nos da el siguiente resultado.

$$\text{Valor Digital} = \frac{1.8V}{3.5V} * (2^{12} - 1)$$

$$\text{Valor Digital} = 2106 \text{ bits}$$

Luego mediante la función mapFloat remapeamos los valores que recibimos desde el sensor a una escala entre 1 y 5, siendo 1 equivalente a los 2106 bits, medidos en agua limpia y 5 equivalente a 0 bits, para agua completamente turbia.

**Figura 35**

*Test del control del sensor de turbidez de agua con la placa ESP32*



```
esp32_sensor_turbidez Arduino 1.8.9
Archivo Editor Programa Herramientas Ayuda

esp32_sensor_turbidez
#define sensorPin 36

int sensorValue = 0;
float turbidity = 0;

void setup() {
  Serial.begin(9600);
  pinMode(sensorPin, INPUT);
}

void loop() {
  sensorValue = analogRead(sensorPin);

  turbidity = mapFloat(sensorValue, 0.0, 2100.0, 5.0, 1.0);

  Serial.println(turbidity);

  delay(1000);
}

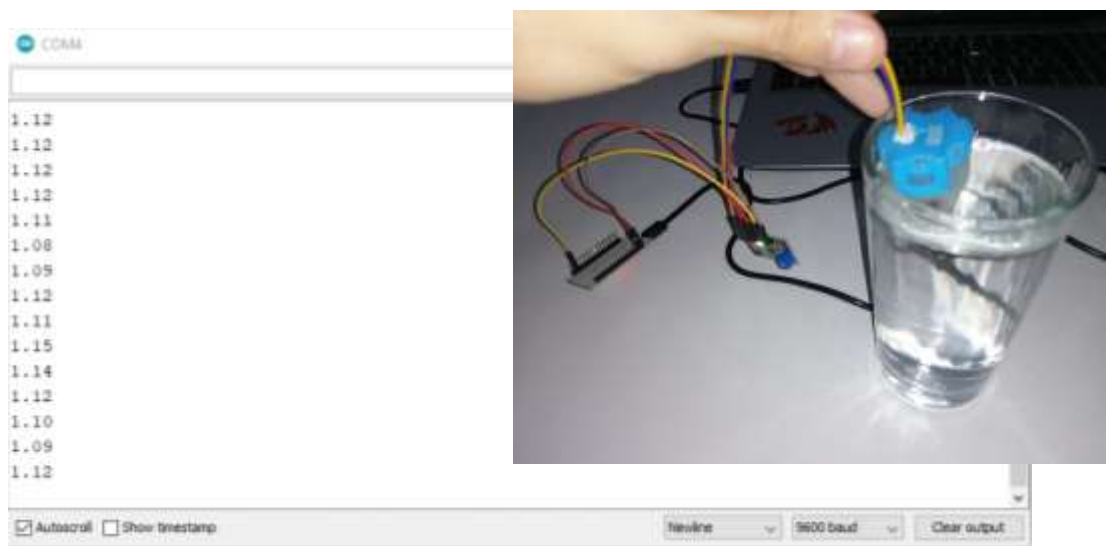
float mapFloat(int x, float in_min, float in_max, float out_min, float out_max)
{
  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
}
```

**Nota.** Codificación del test del control del sensor de turbidez de agua con la placa ESP32. Elaboración propia.

La escala de turbidez que le hemos considerado en la función `mapFloat` es de 1.0 cuando el agua está limpia y de 5.0 cuando está completamente turbia. Teniendo en cuenta que el voltaje máximo que mide el ADC cuando el agua está limpia es de 1.8 V DC y cuando está sucia, mayor de 1.8 V DC, y nuestro ADC lo convierte en bits.

**Figura 36**

*Sensor de Turbidez conectado a la placa ESP32, sumergido en agua limpia*



**Nota.** Sensor de Turbidez conectado a la placa ESP32, sumergido en agua limpia, en el monitor serie podemos ver los valores. Elaboración propia.

**Figura 37**

*Sensor de Turbidez conectado a la placa ESP32, sumergido en agua turbia*



**Nota.** Sensor de Turbidez conectado a la placa ESP32, sumergido en agua turbia, en el monitor serie podemos ver los valores. Elaboración propia.



#### 3.4.1.4. Sensor Ultrasónico

Este dispositivo que tiene la capacidad de medir la distancia de un objeto a través del sonido, utilizando sus ondas ultrasónicas para efectuar su funcionamiento, para ello hemos definido las constantes trigPin, echoPin y la velocidad del sonido. Luego en el en el setup, se inicializa trigPin el cual hace referencia al pin 5 inicializado como OUTPUT (salida) y el echoPin hace referencia al pin 18 inicializado como INPUT (entrada), de la placa ESP32.

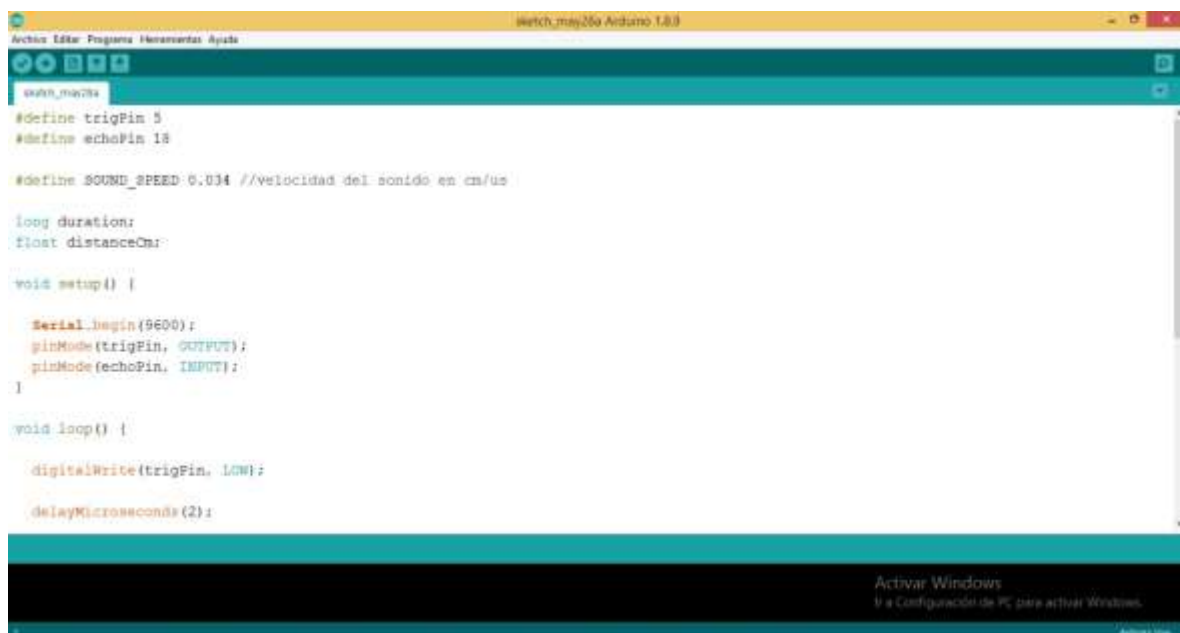
Luego en el loop, se envía un 0 (LOW) por el pin TRIG del sensor, por 2 milisegundos, luego un 1 (HIGH) por 10 milisegundos, para activar el sensor ultrasónico, y luego un 0 (LOW) otra vez para apagar el envío de la onda de sonido del sensor, una vez que la onda de sonido ha regresado, el sensor ultrasónico mediante la función pulseIn, que se encarga de medir la duración de la señal emitida por el Sensor Ultrasónico, emite un pulso mediante el pin ECHO, de duración igual al recorrido que ha hecho la onda de sonido, posteriormente se calcula la medida mediante la fórmula siguiente:

$$Distancia = \frac{(Tiempo\ del\ pulso\ ECO) * (Velocidad\ del\ sonido = 340m/s)}{2}$$

Ya que la onda de sonido ha recorrido el doble del camino al ir y regresar al sensor.

**Figura 38**

*Sensor Ultrasónico conectado a la placa ESP32, parte 1*



**Nota.** Sensor Ultrasónico conectado a la placa ESP32, parte 1. Elaboración propia.

**Figura 39**

*Sensor Ultrasónico conectado a la placa ESP32, parte 2*



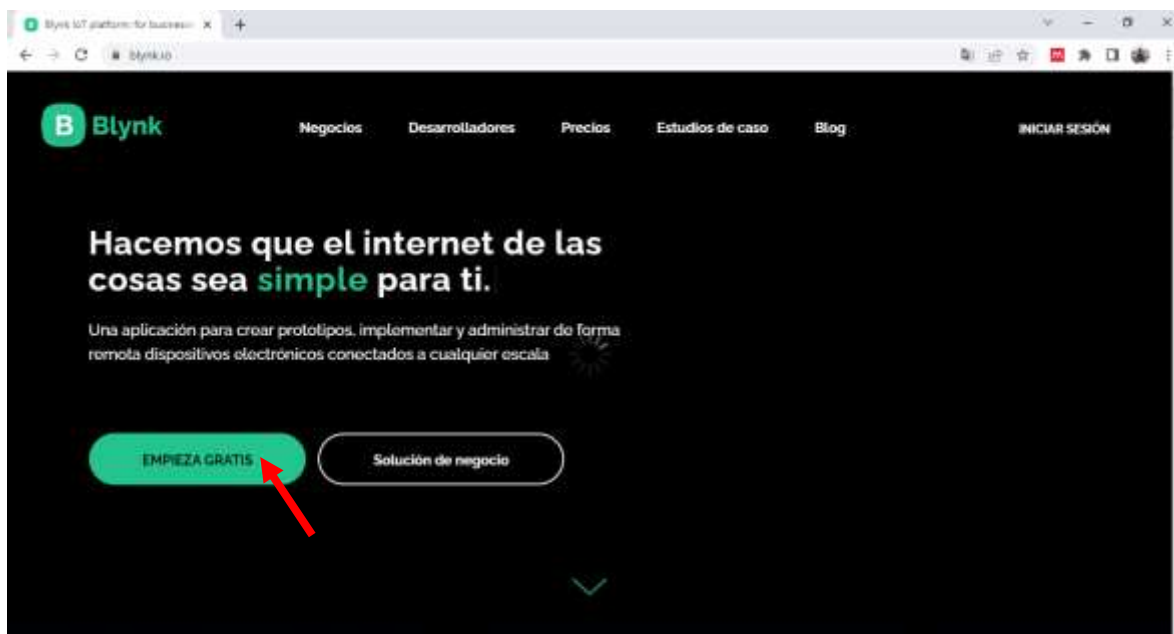
**Nota.** Sensor Ultrasónico conectado a la placa ESP32, parte 2. Elaboración propia.

### 3.4.2. Conexión de la App Blynk y el sistema

Lo primero que haremos es crear una cuenta en la web de Blynk, entrando al siguiente enlace <https://blynk.io/>, donde nos mostrará la ventana principal de la web y haremos clic en el botón “EMPEZAR GRATIS”.

**Figura 40**

*Ventana principal de la página de App Blynk.*

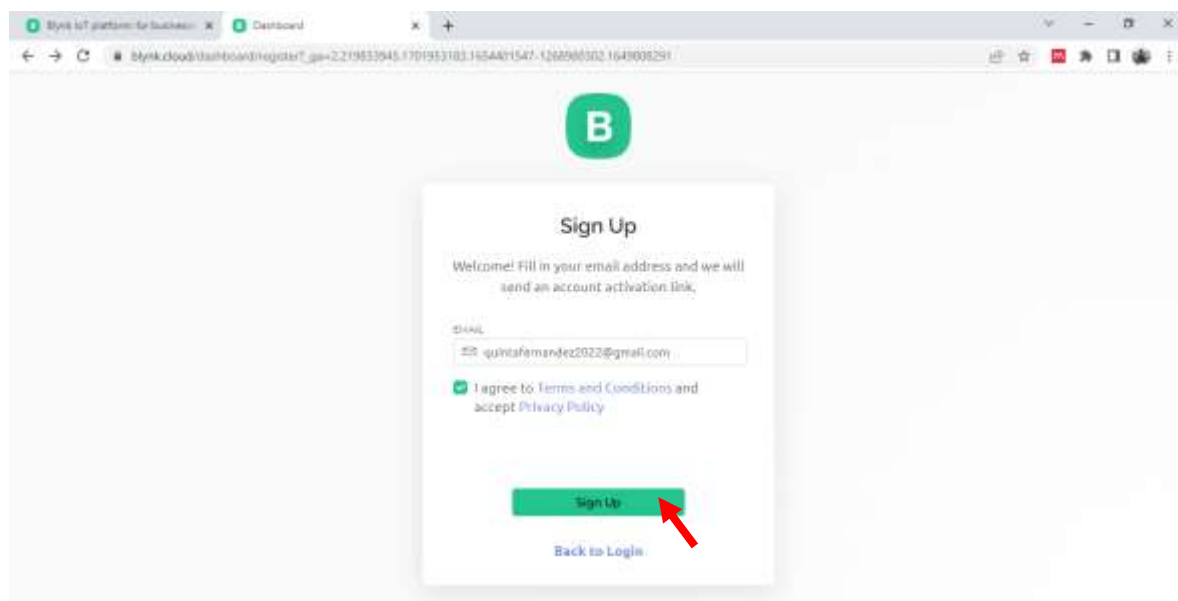


**Nota.** Ventana principal de la página de App Blynk. Tomada de la página (Blynk, n.d.)

Se mostrará la siguiente pantalla, pondremos nuestro correo, aceptamos los términos y condiciones, luego haremos clic en “Sign Up” para inscribirnos.

**Figura 41**

*Ventana para crear una cuenta de la App Blynk*

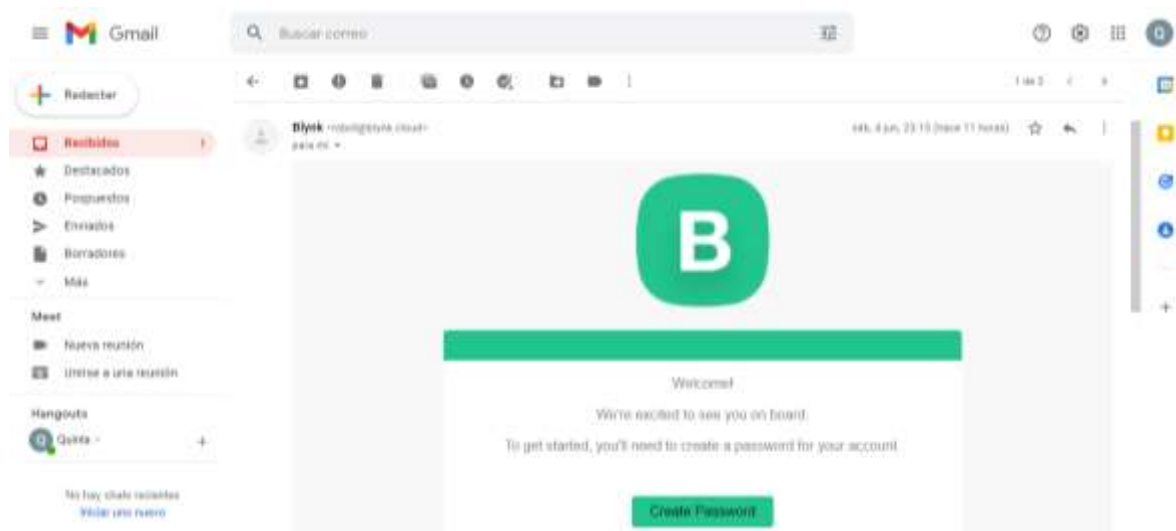


**Nota.** Ventana para crear una cuenta de la App Blynk. Tomada de la página (Blynk, n.d.)

En seguida nos llegará un correo electrónico con el enlace de activación de la cuenta.

**Figura 42**

*Correo de activación*

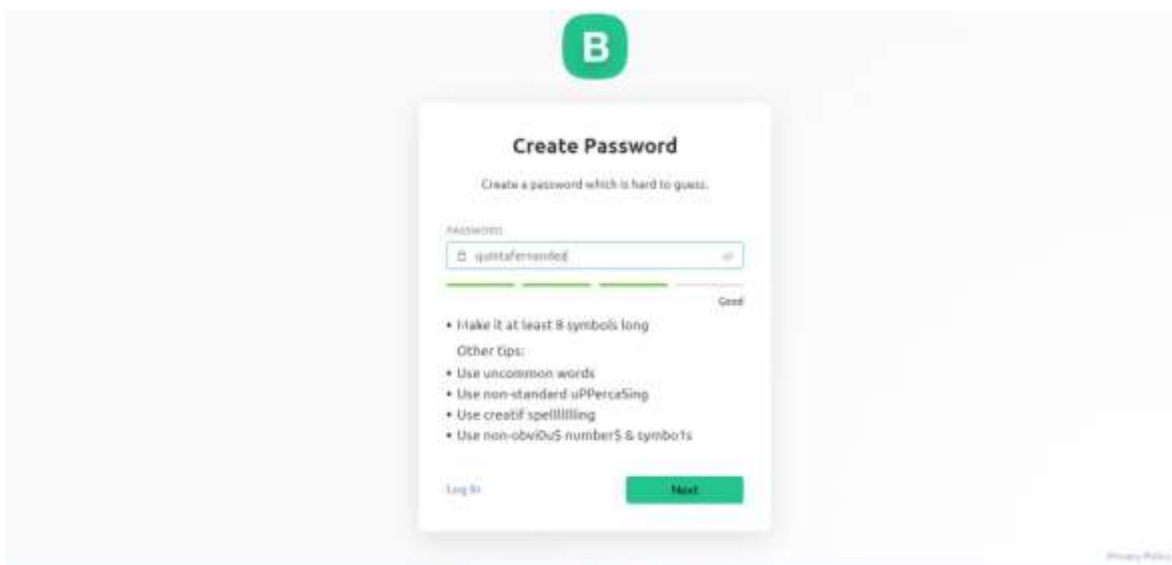


**Nota.** Recepción del correo de activación. Tomada de correo personal

Procedemos a crear nuestra contraseña.

**Figura 43**

*Creación de contraseña de Blynk.*

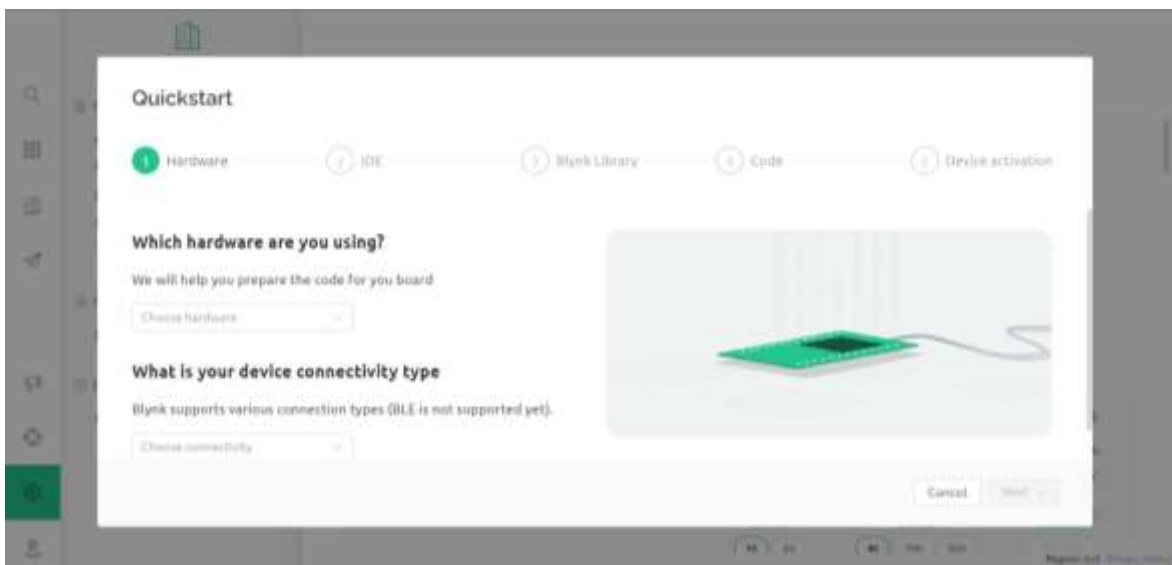


**Nota.** Creación de contraseña de Blynk. Tomada de la página (Blynk, n.d.)

Le damos “Next” a todo y al final nos mostrará la siguiente ventana, hacemos clic en “cancel”

**Figura 44**

*Ventana de Blynk.*

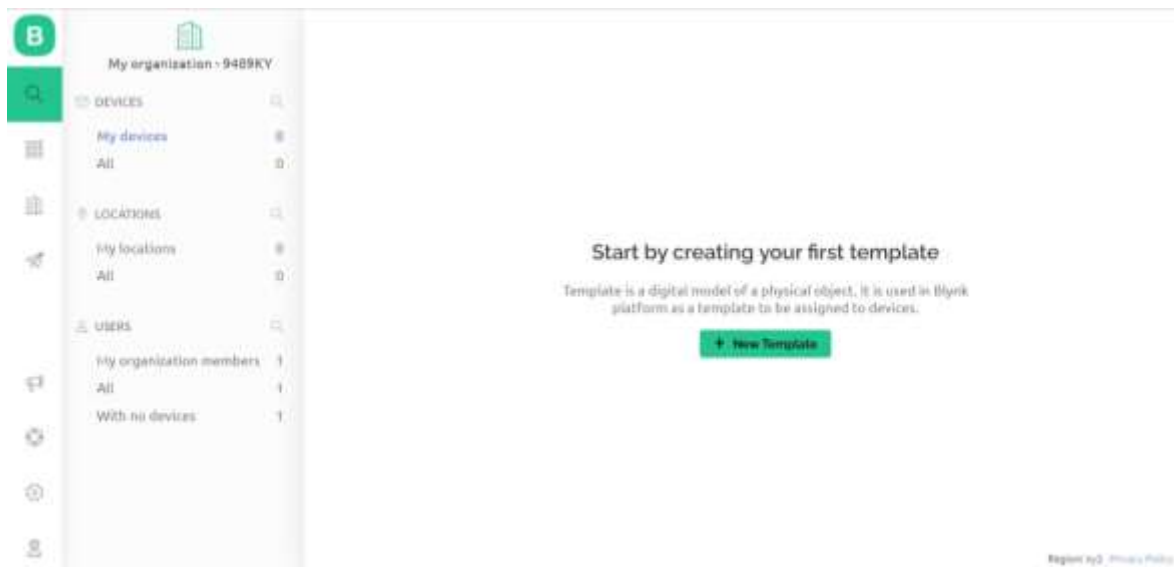


**Nota.** Ventana de Blynk. Tomada de la página (Blynk, n.d.)

Y ahora si ya podemos empezar a crear nuestra plantilla.

**Figura 45**

*Creación de plantillas de Blynk, parte 1*

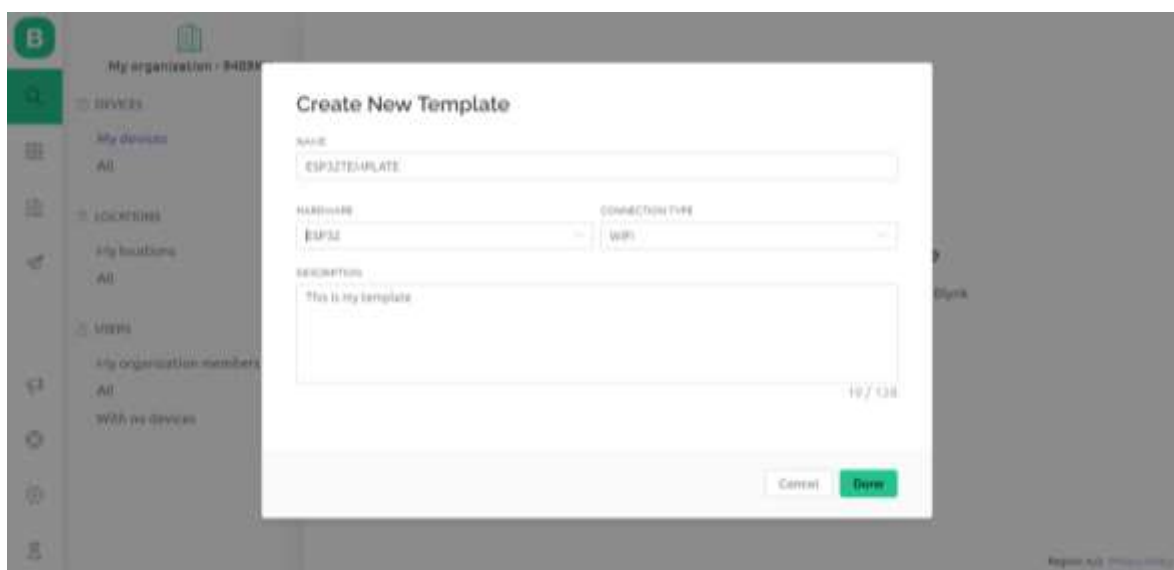


**Nota.** Primera parte de la creación de plantillas de Blynk. Tomada de la página (Blynk, n.d.)

Seleccionamos el dispositivo y el tipo de conexión que usaremos.

**Figura 46**

*Creación de plantillas de Blynk, parte 2*



**Nota.** Segunda parte de la creación de plantillas de Blynk. Tomada de la página (Blynk, n.d.)

Procedemos a guardar la plantilla.

**Figura 47**

*Guardar plantilla.*

The screenshot shows the 'ESP32TEMPLATE' configuration page in the Blynk web interface. The page has a sidebar with navigation icons and a main content area. The main content area includes tabs for 'Info', 'Metadata', 'Datastreams', 'Events', 'Automations', 'Web Dashboard', and 'Mobile Dashboard'. The 'Info' tab is active, showing fields for 'TEMPLATE NAME' (ESP32TEMPLATE), 'HARDWARE' (ESP32), 'CONNECTION TYPE' (WiFi), 'DESCRIPTION' (This is my template), 'TEMPLATE ID' (ESP32TEMPLATE), 'MANUFACTURER' (My organization - 9488KY), 'OFFLINE IGNORE PERIOD' (00:00:00 - 00:00:00), and 'HOTSPOT NAME'. There is also a section for 'TEMPLATE IMAGE (OPTIONAL)' with an 'Add image' button and a note about image requirements. A 'FIRMWARE CONFIGURATION' section shows a code snippet for 'blynk\_template\_id' and 'blynk\_device\_name'. A 'Save' button is in the top right corner.

**Nota.** Guardar plantilla. Tomada de la página (Blynk, n.d.)

Hacemos clic en “New Device”, para crear un nuevo dispositivo.

**Figura 48**

*Crear dispositivo*

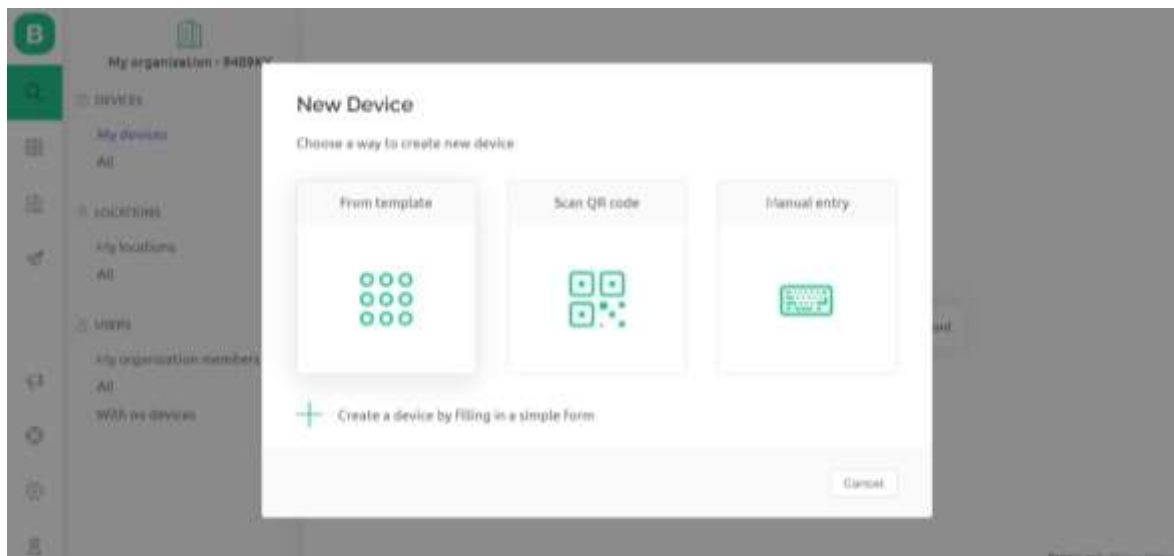
The screenshot shows the 'My organisation - 9488KY' page in the Blynk web interface. The page has a sidebar with navigation icons and a main content area. The main content area includes a list of 'DEVICES' (My devices, All), 'LOCATIONS' (My locations, All), and 'USERS' (My organisation members, All, With no devices). There is a large section titled 'All of your devices will be here.' with a note about activating new devices by using the app for iOS or Android. Below this are buttons for 'Download for iOS' and 'Download for Android', and a prominent green '+ New Device' button.

**Nota.** Crear dispositivo. Tomada de la página (Blynk, n.d.)

Luego elegimos el dispositivo, en nuestro caso optamos por “From Template”.

**Figura 49**

*Elección de dispositivos.*

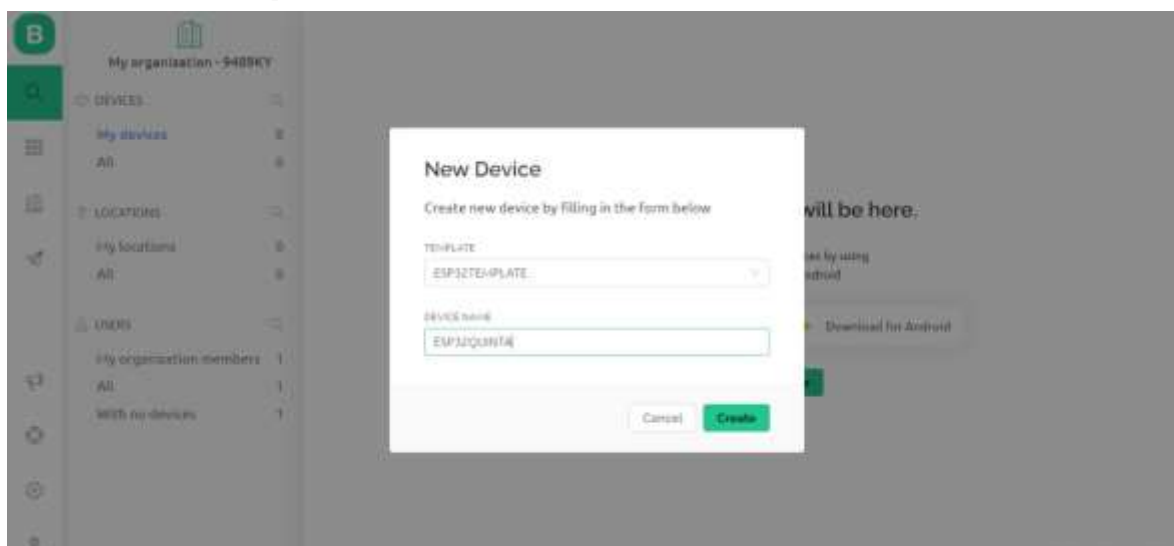


**Nota.** Elección de dispositivo. Tomada de la página (Blynk, n.d.)

A continuación, enviaremos un formulario para crear un nuevo dispositivo; seleccionamos la plantilla anteriormente creada y el nombre del dispositivo, en nuestro caso le colocamos el nombre de nuestro proyecto.

**Figura 50**

*Creación de un nuevo dispositivo.*



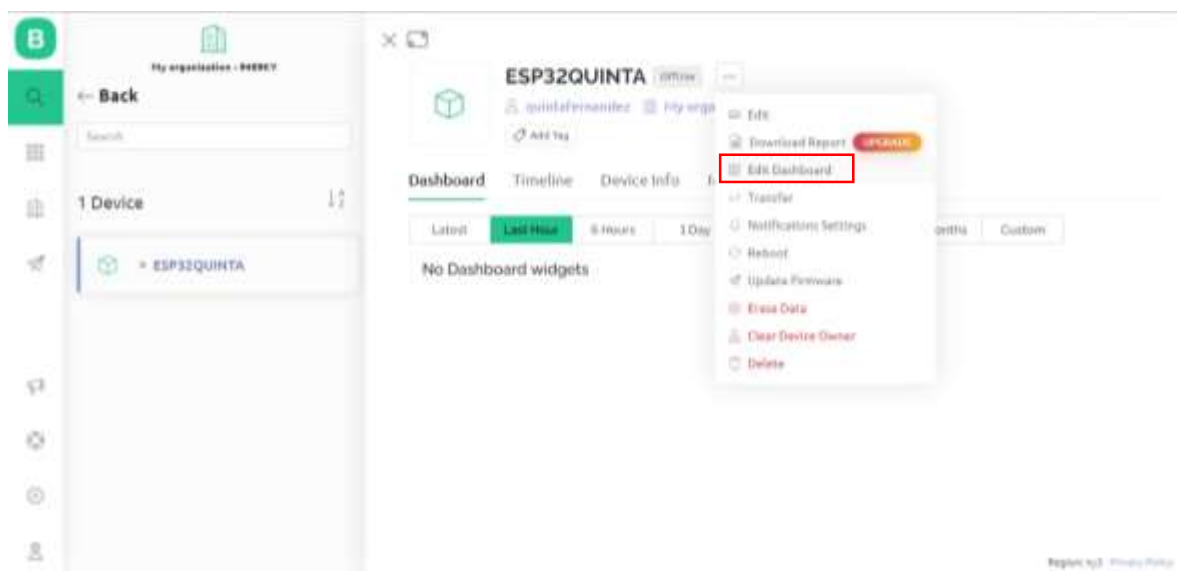
**Nota.** Creación de un dispositivo. Tomada de la página (Blynk, n.d.)



Seleccionamos “Edit Dashboard” para empezar a editar nuestro panel y crear nuestras variables (Datastreams).

**Figura 51**

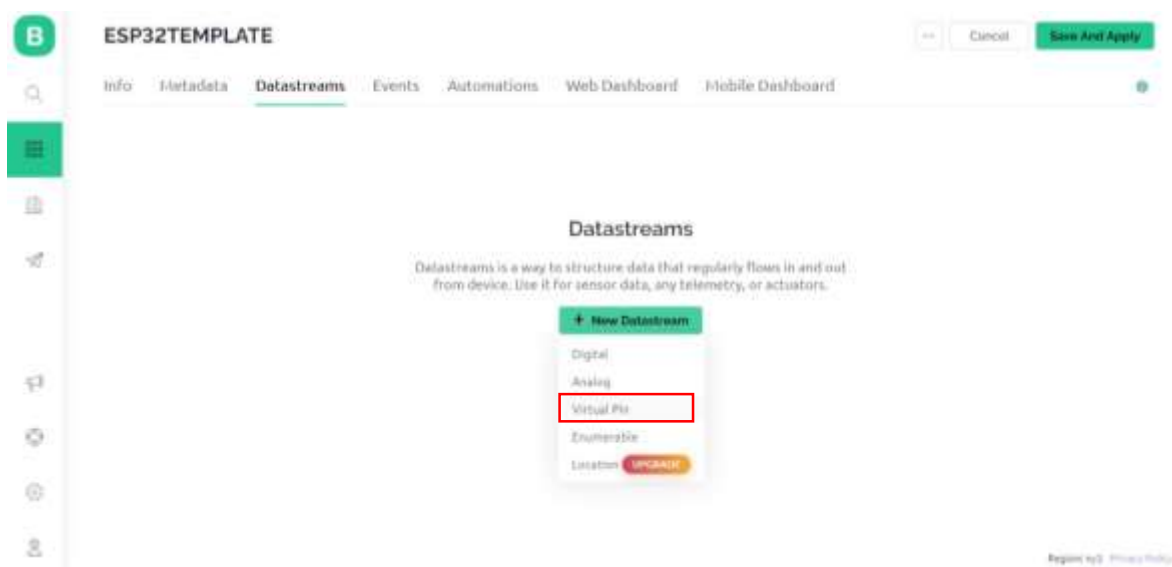
*Edición de panel.*



**Nota.** Edición de panel. Tomada de la página (Blynk, n.d.)

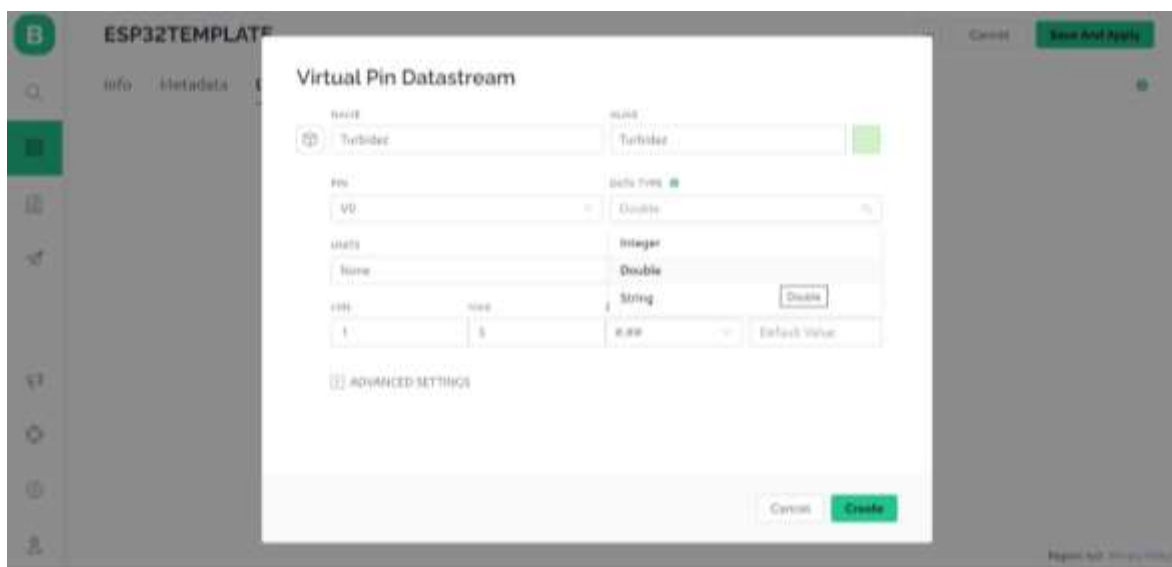
Ahora crearemos nuestra Datastreams (Flujo de datos), que nos permitirá comunicar la App Blynk o servicio de la web de Blynk, con la placa ESP32.

Los Datastreams, son tipos de datos que nosotros mismos vamos a definir y que se van a comunicar a través de los “Virtual Pin”, los cuales vamos a seleccionar para elegir la interfaz con la que se intercambiará el estado de las variables.

**Figura 52***Flujo de datos*

**Nota.** Flujo de datos. Tomada de la página (Blynk, n.d.)

Ahora si procedemos a la creación de variables, primero configuramos el nombre y luego el tipo de dato que recibirá, en este caso para la turbidez, será de tipo Double, ya que recibirá decimales, y el valor mínimo que hemos considerado es 1 y el máximo el 5.

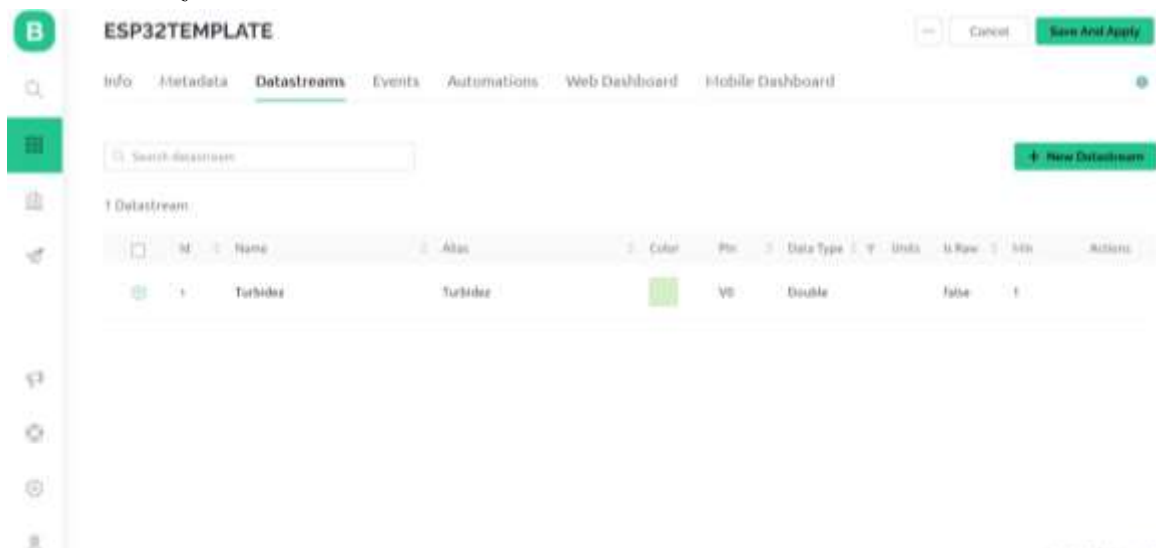
**Figura 53***Creación de la variable Turbidez*

**Nota.** Creación de la variable Turbidez. Tomada de la página (Blynk, n.d.)

Una vez creada nuestra variable Turbidez, se mostrará de la siguiente manera, donde se le asignado el Pin Virtual V<sub>0</sub>.

**Figura 54**

*Variable Turbidez.*

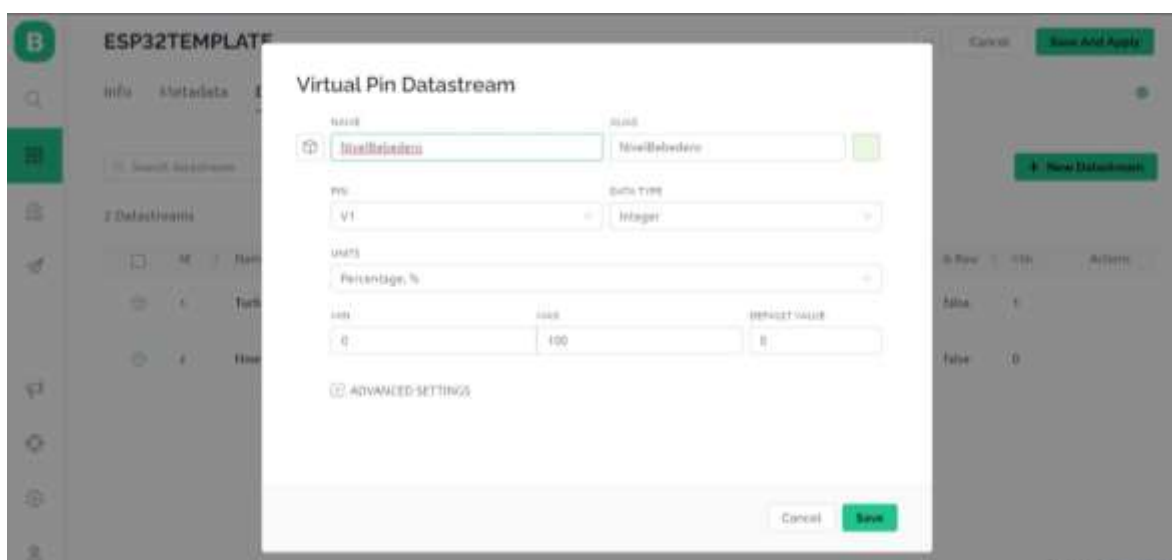


**Nota.** Variable Turbidez. Tomada de la página (Blynk, n.d.)

Ahora vamos a crear la variable NivelBebedero, la cual será de tipo Integer, ya que recibirá valores enteros, y se le asignará el Pin Virtual V<sub>1</sub>.

**Figura 55**

*Creación de la variable NivelBebedero.*



**Nota.** Creación de la variable NivelBebedero. Tomada de la página (Blynk, n.d.)

Posteriormente crearemos la variable VálvulaSuperior, la cual también será de tipo Integer, por que recibirá valores enteros, y se le asignará el Pin Virtual V<sub>2</sub>.

**Figura 56**

*Creación de la variable VálvulaSuperior*

The screenshot shows a 'Virtual Pin Datastream' dialog box. The 'NAME' field contains 'VálvulaSuperior' and the 'ALIAS' field contains 'Vclsup'. The 'PIN' field contains 'V2' and the 'DATA TYPE' field contains 'Integer'. The 'UNIT' field is set to 'None'. Below these fields, there are input fields for 'MIN' (0), 'MAX' (1), and 'DEFAULT VALUE' (0). At the bottom, there is an 'ADVANCED SETTINGS' section which is currently collapsed. The dialog has 'Cancel' and 'Create' buttons at the bottom right.

**Nota.** Creación de la variable VálvulaSuperior. Tomada de lag página (Blynk, n.d.)

Luego crearemos la variable VálvulaInferior, la cual también será de tipo Integer, por que recibirá valores enteros, y se le asignará el Pin Virtual V<sub>3</sub>.

**Figura 57**

*Creación de la variable VálvulaInferior*

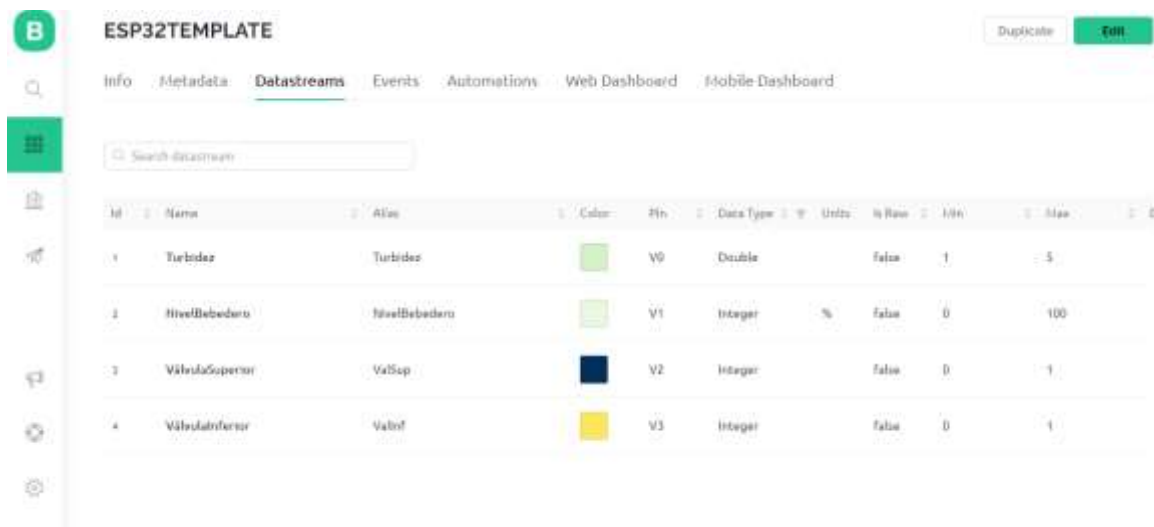
The screenshot shows a 'Virtual Pin Datastream' dialog box. The 'NAME' field contains 'VálvulaInferior' and the 'ALIAS' field contains 'Vclinf'. The 'PIN' field contains 'V3' and the 'DATA TYPE' field contains 'Integer'. The 'UNIT' field is set to 'None'. Below these fields, there are input fields for 'MIN' (0), 'MAX' (1), and 'DEFAULT VALUE' (0). At the bottom, there is an 'ADVANCED SETTINGS' section which is currently collapsed. The dialog has 'Cancel' and 'Create' buttons at the bottom right.

**Nota.** Creación de la variable VálvulaInferior. Tomada de lag página (Blynk, n.d.)

Finalmente guardamos los cambios y verificamos nuestras variables creadas.

**Figura 58**

*Variables Creadas en la Datastream*



The screenshot shows the Blynk ESP32TEMPLATE interface with the 'Datastreams' tab selected. A table lists four created variables:

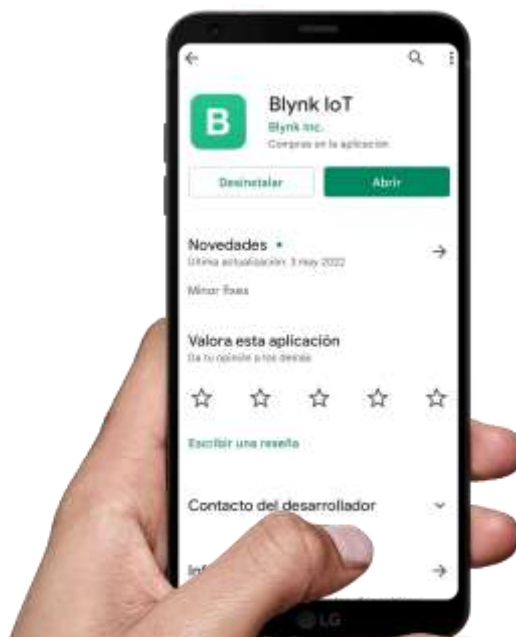
ID	Name	Alias	Color	Pin	Data Type	Units	Is Raw	Min	Max
1	Turbidez	Turbidez	Light Green	V0	Double		false	1	5
2	NivelBebadero	NivelBebadero	Light Green	V1	Integer	%	false	0	100
3	ValvulaSuperior	ValSup	Dark Blue	V2	Integer		false	0	1
4	ValvulaInferior	ValInf	Yellow	V3	Integer		false	0	1

**Nota.** Variables Creadas en la Datastream. Tomada de la página (Blynk, n.d.)

Una vez creadas todas nuestras variables, procederemos a descargar la App Blynk de la Play Store en nuestro teléfono.

**Figura 59**

*Descarga de la App Blynk*



**Nota.** Variables Creadas en la Datastream. Tomada de la aplicación (Play Store)

Abrimos la App Blynk y hacemos clic en “Sign Up”.

**Figura 60**

*Inicia sesión de la App Blynk*

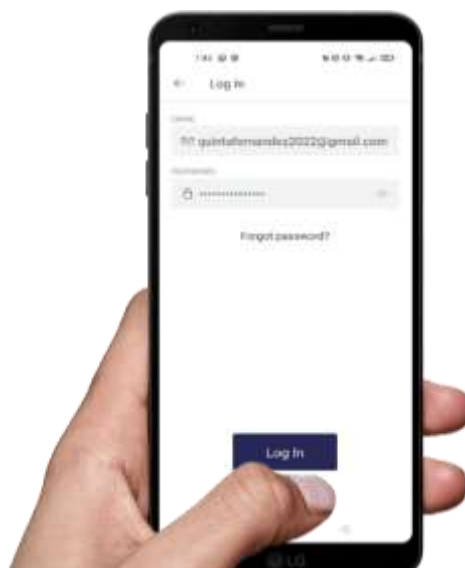


**Nota.** Iniciar sesión de la App Blynk. Tomada de la aplicación (App Blynk)

Aquí llenaremos los campos solicitados para loguearse y haremos clic en “Log In”.

**Figura 61**

*Logueo de la App Blynk*

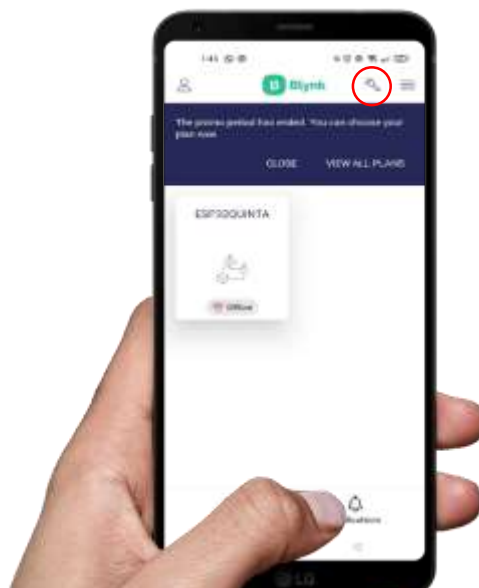


**Nota.** Logueo de la App Blynk. Tomada de la aplicación (App Blynk)

Nos mostrará el dispositivo que creamos en la web de Blynk, luego haremos clic en la llave que se encuentra en la parte superior.

### Figura 62

*Dispositivo creado en la web de Blynk*



**Nota.** Dispositivo creado en la web de Blynk. Tomada de la aplicación (App Blynk)

Después nos mostrará la ventana en modo desarrollador y haremos clic en “Prototype”

### Figura 63

*Ventana en modo desarrollo de la App Blynk*

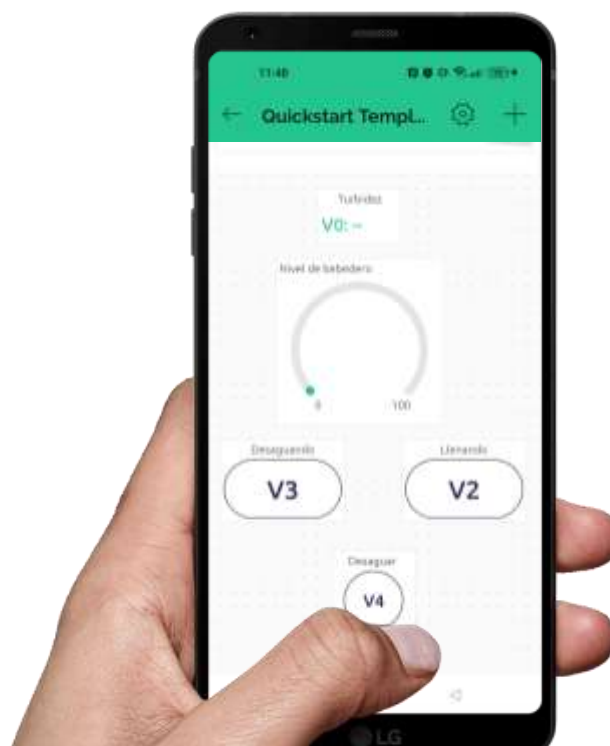


**Nota.** Ventana en modo desarrollo de la App Blynk. Tomada de la aplicación (App Blynk)

Ahora si procedemos a la creación de botones y diseño de interfaces, en esta ventana podremos agregar el widget que necesitamos para mostrar los valores de los sensores.

**Figura 64**

*Creación de botones en la App Blynk*



**Nota.** Creación de botones en la App Blynk. Tomada de la aplicación (App Blynk)



Finalmente nos quedará de la siguiente manera:

**Figura 65**

*Interfaz de Usuario*



**Nota.** El usuario verá esta interfaz al entrar a la App Blynk. Tomada de la aplicación (App Blynk)

### 3.4.3. Codificación del sistema

Para empezar con el desarrollo del código primero debemos instalar las librerías de Blynk e instalar el paquete de configuración de tarjetas ESP32 en el IDE Arduino.

Nosotros lo descargamos de los siguientes sitios web:

Librerías de Blynk: <https://docs.blynk.io/en/blynk.edgent-firmware-api/supported-boards>

Paquete de configuración de tarjetas ESP32: <https://docs.espressif.com/projects/arduino-esp32/en/latest/installing.html#installing-using-arduino-ide>

Luego de descargar las librerías de Blynk, automáticamente al seleccionarlás se nos crearan las credenciales de Blynk en nuestro proyecto.

#### Figura 66

*Credenciales de Blynk*

```
1 #define BLYNK_TEMPLATE_ID "TMPLcHZdoVlD"
2 #define BLYNK_DEVICE_NAME "Quickstart Template"
3 #define BLYNK_AUTH_TOKEN "oqFzdRilCH3TlgXOG-tbfDnUeXvNVi0M"
4
5 #define BLYNK_FIRMWARE_VERSION "0.1.0"
6
7 #define BLYNK_PRINT Serial
8 #define APP_DEBUG
9
10 #include "BlynkEdgent.h"
```

**Nota.** Credenciales de Blynk. Elaboración propia.

Después agregamos la biblioteca ESP\_Mail\_Client que nos permitirá realizar el envío de correos electrónicos.

#### Figura 67

*Inclusión de la biblioteca ESP\_Mail\_Client*

```
12 #include <ESP_Mail_Client.h>
```

**Nota.** Inclusión de la biblioteca ESP\_Mail\_Client para envío de correos electrónicos. Elaboración propia.

Continuamos con la definición de variables, el correo que se ha usado para el envío y la recepción es el mismo.

### Figura 68

*Definición de constantes para el envío de correo.*

```
14 #define SMTP_HOST "smtp.gmail.com"
15 #define SMTP_PORT 465
16
17 #define AUTHOR_EMAIL "quintafernandez2022@gmail.com"
18 #define AUTHOR_PASSWORD "lynqscftvrmorgys"
19
20 #define RECIPIENT_EMAIL "quintafernandez2022@gmail.com"
21 SMTPSession smtp;
```

**Nota.** Definición de constantes para el envío de correo electrónico al ganadero. Elaboración propia.

Ahora crearemos la función `void smtpCallback(SMTP_Status status);` que nos permitirá realizar la operación de envío de correos.

### Figura 69

*Creación de función para envío de correos.*

```
23 void smtpCallback(SMTP_Status status);
```

**Nota.** Creación de función para el envío de correo electrónico al ganadero. Elaboración propia.

Posteriormente definimos las constantes de los pines que usaremos

**Figura 70**

*Definición de constantes de los pines.*

```

25 #define trigPin 5
26 #define echoPin 18
27 #define SOUND_SPEED 0.034 //velocidad del sonido en cm/us
28
29 #define turbiditySensorPin 36
30
31 #define valvePinSup 23
32 #define valvePinInf 22
33
34 #define levelSensorPinSup 32
35 #define levelSensorPinInf 12
36
37 #define stopButtonPin 4
38
39 #define valveInfButtonPin 33

```

**Nota.** Definición de constantes que indicarán los pines a usar del ESP32. Elaboración propia.

Ahora creamos la función setup y dentro de ella declaramos las siguientes variables:

**Figura 71**

*Declaración de Variables*

```

43 int levelSensorSupState;
44 int levelSensorInfState;
45
46 int valveSupState = 0;
47 int valveInfState = 0;
48
49 int stopButtonState = 0;
50 int valveInfButtonState = 0;
51 int virtualValveInfButtonState = 0;
52
53 int sensorSupOldState = 0;
54
55 int oldTurbidityState = 0;
56 int newTurbidityState = 0;
57
58 float waterTurbidity;

```

**Nota.** Declaración de variables globales. Elaboración propia.

También, declaramos como entradas y salidas los pines donde se conectarán los sensores, botones pulsadores y relé.

**Figura 72**

*Declaración de pines.*

```

66  pinMode(trigPin, OUTPUT);
67  pinMode(echoPin, INPUT);
68
69  pinMode(turbiditySensorPin, INPUT);
70
71  pinMode(valvePinSup, OUTPUT);
72  pinMode(valvePinInf, OUTPUT);
73
74  pinMode(levelSensorPinInf, INPUT_PULLUP);
75  pinMode(levelSensorPinSup, INPUT_PULLUP);
76
77  pinMode(stopButtonPin, INPUT_PULLUP);
78  pinMode(valveInfButtonPin, INPUT_PULLUP);
79
80  BlynkEdgent.begin();

```

**Nota.** Declaración de entradas y salidas de los pines donde se conectarán los periféricos. Elaboración propia.

Y finalmente establecemos los tiempos y el intervalo en el que se ejecutarán las funciones.

**Figura 73**

*Funciones e Intervalos de tiempo*

```

83  timer.setInterval(1000L, sendWaterLevel);
84  timer.setInterval(2000L, sendTurbidity);
85  timer.setInterval(500L, sendValveState);

```

**Nota.** Tiempos e intervalo en el que se ejecutarán las funciones. Elaboración propia.

Y en nuestro bucle loop leemos el estado del botón de parada, sensores de boya botón de desagüe y las válvulas.

**Figura 74***Lectura de variables*

```

88 void loop() {
89
90     stopButtonState = digitalRead(stopButtonPin);
91
92     levelSensorSupState = digitalRead(levelSensorPinSup);
93     levelSensorInfState = digitalRead(levelSensorPinInf);
94
95     valveInfButtonState = digitalRead(valveInfButtonPin);
96
97     valveSupState = digitalRead(valvePinSup);
98     valveInfState = digitalRead(valvePinInf);
99
100    BlynkEdgent.run();
101
102    timer.run();
103 }

```

**Nota.** Lectura de variables en el loop. Elaboración propia.

Ahora crearemos nuestras funciones:

- Creamos la función `sendWaterLevel` la que nos permitirá enviar el nivel de agua actual al servidor de Blynk, es por ello que dentro de esta ejecutamos la función `measureWaterLevel` para obtener el nivel de llenado del Bebedero y luego actualizamos el valor del pin virtual `V1` en Blynk con el nivel del agua (`waterLevel`).

**Figura 75***Función `sendWaterLevel`*

```

105 void sendWaterLevel() {
106     //Ejecutamos la función measureWaterLevel para obtener el nivel de llenado
107     //del Bebedero mediante el sensor ultrasónico
108     int waterLevel = measureWaterLevel();
109
110     //Actualizamos el Valor del pin virtual V1 en Blynk con el nivel del agua
111     Blynk.virtualWrite(V1, waterLevel);
112 }

```

**Nota.** Función `sendWaterLevel`. Elaboración propia.

- Para determinar en la aplicación si el agua está sucia creamos la función `sendTurbidity` en la cual ejecutaremos la función `measureTurbidity` para obtener el valor de la Turbidez del agua desde el sensor. Después actualizamos el valor del pin virtual `V0` en Blynk con el valor obtenido.

### Figura 76

*Función `sendTurbidity`*

```

114 void sendTurbidity() {
115     //Ejecutamos la función measureTurbidity para obtener el valor de la Turbidez de agua
116     waterTurbidity = measureTurbidity();
117
118     //Actualizamos el Valor del pin virtual V0 en Blynk con valor de la turbidez del agua
119     Blynk.virtualWrite(V0, waterTurbidity);
120 }
```

**Nota.** Función `sendTurbidity`. Elaboración propia.

- La función `sendValveState` nos permite controlar y enviar el estado de las válvulas, tanto la inferior como la superior. Estas válvulas controlan el flujo del agua.

Mediante la ESP32, nosotros las podemos abrir y cerrar las válvulas como caños. Para ello debemos ejecutar la función `valveControl`, la cual nos permitirá controlar las válvulas, luego mediante condicionales cambiar el color del indicador de las válvulas de acuerdo a su estado y finalmente actualizaremos su estado en Blynk.

**Figura 77***Función sendValveState*

```

122 void sendValveState() {
123     //Ejecutamos la función valveControl para controlar automáticamente las válvulas
124     valveControl();
125     //Cambiamos el color del indicador en las válvulas en Blynk dependiendo de su estado
126     if (valveSupState) {
127         Blynk.setProperty(V2, "color", "#CC0000");
128     }
129     else {
130         Blynk.setProperty(V2, "color", "#000000");
131     }
132
133     if (valveInfState) {
134         Blynk.setProperty(V3, "color", "#CC0000");
135     }
136     else {
137         Blynk.setProperty(V3, "color", "#000000");
138     }
139
140     //Actualizamos el estado de las válvulas en Blynk
141     Blynk.virtualWrite(V2, valveSupState);
142     Blynk.virtualWrite(V3, valveInfState);
143 }

```

**Nota.** Función sendValveState. Elaboración propia.

Posteriormente se lee el estado del botón virtual de desagüe de la aplicación Blynk y lo guarda en la variable virtualValveInfButtonState.

**Figura 78***Lectura del estado del botón de desagüe.*

```

145 BLYNK_WRITE(V4) {
146
147     virtualValveInfButtonState = param.asInt();
148
149 }

```

**Nota.** Lee el estado del botón virtual de desagüe de la aplicación Blynk. Elaboración propia.

- La función valveControl, nos permite tener el control total de las válvulas, en el caso el botón de parada se haya presionado, ambas válvulas se desactivarán; si el botón físico o virtual de desagüe se encuentra presionado, la válvula de desagüe se abrirá y la válvula de



llenado se apagará si el sensor de boya superior se ha activado o si la turbidez es mayor o igual a 3 o si la válvula inferior se encuentra abierta. Además, para evitar el funcionamiento intermitente de la válvula de llenado esta solo se abrirá en caso el tanque se encuentre en su nivel mínimo es decir cuando el sensor de nivel inferior se encuentra desactivado.

**Figura 79**

*Función valveControl*

```

150 void valveControl() {
151     int notValveInfButtonState = !valveInfButtonState;
152     int notLevelSensorInfState = !levelSensorInfState;
153     int notLevelSensorSupState = !levelSensorSupState;
154
155     if (!stopButtonState == 1) {
156         digitalWrite(valvePinSup, LOW);
157         digitalWrite(valvePinInf, LOW);
158     } else {
159         digitalWrite(valvePinInf, notValveInfButtonState || virtualValveInfButtonState);
160         if (levelSensorInfState == 0) {
161             sensorSupOldState = 0;
162         }
163         if (notLevelSensorSupState == 1) {
164             sensorSupOldState = 1;
165         }
166         if (waterTurbidity >= 3 || valveInfState) {
167             digitalWrite(valvePinSup, LOW);
168         } else {
169             if (sensorSupOldState == 0 && levelSensorInfState == 0) {
170                 digitalWrite(valvePinSup, HIGH);
171             }
172             if (sensorSupOldState == 0 && levelSensorInfState == 1) {
173                 digitalWrite(valvePinSup, HIGH);
174             }
175             if (sensorSupOldState == 1 && levelSensorInfState == 0) {
176                 digitalWrite(valvePinSup, LOW);
177             }
178             if (sensorSupOldState == 1 && levelSensorInfState == 1) {
179                 digitalWrite(valvePinSup, LOW);
180             }
181         }
182     }
183 }

```

**Nota.** Función valveControl. Elaboración propia.

- Creamos la función `measureTurbidity` para obtener el valor de la Turbidez del agua, esta función nos permitirá obtener el valor de la turbidez, almacenando la medición en una escala de 1 a 5 en la variable `turbidity`, en el caso de que esta sea igual o mayor a 3, se enviará un correo de alerta siempre y cuando anteriormente se encontrara en un valor menor, para no saturar de alertas al ganadero.

**Figura 80***Función `measureTurbidity`*

```

200 float measureTurbidity() {
201     int sensorValue;
202     float turbidity;
203
204     sensorValue = analogRead(turbiditySensorPin);
205     turbidity = mapFloat(sensorValue, 0.0, 2106.0, 5.0, 1.0);
206
207     // Se actualiza el estado de la turbidez, almacenándolo para la próxima medición
208     oldTurbidityState = newTurbidityState;
209
210     // Se modifica la turbidez actual, si esta ha superado o se encuentra en un valor de 3
211     if (turbidity >= 3) {
212         newTurbidityState = 1;
213     } else {
214         newTurbidityState = 0;
215     }
216
217     // Envía un correo de alerta si la turbidez se encuentra o ha superado el valor de 3, solo si
218     // anteriormente se encontraba en un valor menor, para no llenar de alertas al
219     if (oldTurbidityState == 0 && newTurbidityState == 1) {
220         sendMail();
221     }
222     return turbidity;
223 }

```

**Nota.** Función `measureTurbidity`. Elaboración propia.

- Ahora creamos la función `measureWaterLevel`, que nos sirve para medir el nivel del agua en función de porcentaje de llenado, mediante el uso del sensor ultrasónico, transformando las medidas de este siendo el 0% del nivel, 27.1 y 100%, siendo 10.6 centímetros.

**Figura 81***Función measureWaterLevel*

```

225 int measureWaterLevel() {
226     long duration;
227     int levelPercentage;
228     float distanceCm;
229
230     digitalWrite(trigPin, LOW);
231
232     delayMicroseconds(2);
233
234     digitalWrite(trigPin, HIGH);
235
236     delayMicroseconds(10);
237
238     digitalWrite(trigPin, LOW);
239
240     duration = pulseIn(echoPin, HIGH);
241
242     distanceCm = duration * SOUND_SPEED / 2;
243     levelPercentage = map(distanceCm, 10.6, 27.10, 100, 0);
244
245     delay(100);
246
247     return levelPercentage;
248 }

```

**Nota.** Función measureWaterLevel. Elaboración propia.

- Ahora creamos la función mapFloat, que nos sirve para el remapeo o cambio de escala de un valor.

**Figura 82***Función mapFloat*

```

250 float mapFloat(int x, float in_min, float in_max, float out_min, float out_max)
251 {
252     return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
253 }

```

**Nota.** Función mapFloat. Elaboración propia.

Finalmente creamos la función `sendMail`, en la que configuraremos el correo de donde se envía la alerta, además asignamos el mensaje que se enviará.

### Figura 83

#### *Función `sendMail`*

```

255 void sendMail() {
256     smtp.debug(1);
257
258     /* Declaración de sesión de configuración data */
259     ESP_Mail_Session session;
260
261     /* Configuramos el correo y la contraseña de la cuenta de correo de la cual vamos a enviar la alerta */
262     session.server.host_name = SMTP_HOST;
263     session.server.port = SMTP_PORT;
264     session.login.email = AUTHOR_EMAIL;
265     session.login.password = AUTHOR_PASSWORD;
266     session.login.user_domain = "";
267
268     SMTP_Message message;
269
270     /* Configuramos los campos del correo */
271     message.sender.name = "ESP";
272     message.sender.email = AUTHOR_EMAIL;
273     message.subject = "Alerta Bebedero";
274     message.addRecipient("Sara", RECIPIENT_EMAIL);
275
276     String textMsg = "La turbidez del agua ha excedido los valores normales";
277     message.text.content = textMsg.c_str();
278     message.text.charset = "us-ascii";
279     message.text.transfer_encoding = Content_Transfer_Encoding::enc_7bit;
280
281     message.priority = esp_mail_smtp_priority::esp_mail_smtp_priority_low;
282     message.response.notify = esp_mail_smtp_notify_success | esp_mail_smtp_notify_failure | esp_mail_smtp_notify_delay;
283
284     if (!smtp.connect(&session))
285         return;
286
287     if (!MailClient.sendMail(&smtp, &message))
288         Serial.println("Error sending Email, " + smtp.errorReason());
289
290 }
```

**Nota.** Función `sendMail`. Elaboración propia.

#### 3.4.4. Construcción del bebedero a escala

Las medidas reales del bebedero en estudio son de 60 cm de profundidad, 106.5 cm de largo por 106.5 cm de ancho, vamos a construir el prototipo del bebedero aplicando escala de reducción de 1 a 3, para ello haremos el siguiente procedimiento.

- Primera escala para el lado, como nuestro bebedero es cuadrado haremos de un solo lado:

$$Escala = \frac{Medidas\ del\ prototipo}{Medidas\ reales}$$

$$\frac{1}{3} = \frac{Medidas\ del\ prototipo}{105\ cm}$$

$$\frac{106.5\ cm}{3} = Medida\ del\ prototipo$$

$$35.5\ cm = Medida\ del\ prototipo$$

- La segunda escala sería para la profundidad, de igual manera aplicamos la siguiente formula:

$$Escala = \frac{Medidas\ del\ prototipo}{Medidas\ reales}$$

$$\frac{1}{3} = \frac{Medidas\ del\ prototipo}{60\ cm}$$

$$\frac{60\ cm}{3} = Medida\ del\ prototipo$$

$$20\ cm = Medida\ del\ prototipo$$

Las medidas de nuestro bebedero a escala serian 20 cm de profundidad, 35.5 cm de largo y 35.5 cm de ancho.

**Figura 84***Bebedero escala*

**Nota.** Bebedero construido a escala, hecho en vidrio con dimensiones de 35,5cmx35,5cmx20cm. Elaboración propia.

#### ***3.4.5. Ensamblaje o armado de prototipo***

El prototipo está elaborado en material de vidrio con dimensiones de 35,5cmx35,5cmx20cm, asentado sobre una base con un ángulo de inclinación de cinco grados, el recipiente de vidrio consta de 4 orificios para el ensamblado de los sensores. El orificio que se encuentra en la base del bebedero está ubicado en la parte más baja de la base, ya que dicho orificio servirá para el vaciado del bebedero, el cual será controlado por la válvula solenoide de desagüe; el orificio de la parte inferior de una de las caras del bebedero, se encuentra el sensor de turbidez, el cual nos permitirá determinar los valores de la turbidez del agua; en la parte inferior de una cara en la cual no se encuentre otro sensor se ubica el sensor de nivel de líquido inferior que es usado para controlar el nivel mínimo de agua en el que se encuentra el bebedero, en la sección superior de una de las caras del bebedero que no contenga el sensor de turbidez o el sensor de líquido inferior tenemos al sensor de nivel de líquido superior, el cual cumple la función de controlar el nivel máximo del bebedero y finalmente en la parte externa del bebedero se ubicarán la válvula solenoide superior usada para

el llenado del bebedero y el sensor ultrasónico para la visualización del nivel del agua en la aplicación Blynk.

Para mayor información consultar el **Anexo N° 5: Imágenes del montaje del circuito del bebedero automático.**

### 3.5.Despliegue entrega y retroalimentación

#### 3.5.1. Pruebas de Funcionamiento

##### 3.5.1.1. Llenado de Bebedero

El llenado del prototipo de bebedero para ganado, tarda en llenarse tres minutos.

A continuación, se visualizará los valores de los sensores en cada minuto hasta que llegue al nivel máximo de llenado.

**Tabla 12**

*Funcionamiento del llenado de bebedero*

	Tiempo			
	4:00 PM	4:02 PM	4:04 PM	4:06 PM
Sensores	Minuto 0	Minuto 2	Minuto 4	Minuto 6
Sensor de turbidez	1.0	1.0	1.0	1.0
Sensor de nivel de líquido horizontal superior	Boya en la parte inferior	Boya en la parte inferior	Boya en la parte inferior	Boya en la parte superior
Sensor de nivel de líquido horizontal inferior	Boya en la parte inferior	Boya en la parte superior	Boya en la parte superior	Boya en la parte superior
Ultrasónico	0%	33%	66%	100%
Válvula solenoide superior	Válvula abierta	Válvula abierta	Válvula abierta	Válvula cerrada
Válvula solenoide inferior	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula cerrada

**Nota.** Esta tabla muestra cómo funciona el llenado del bebedero.

### 3.5.1.2. Cuando el agua está limpia

Tenemos el bebedero lleno al cien por ciento, simularemos el proceso de que el ganado venga a tomar agua, y para ello se extraerá agua del bebedero hasta que el nivel de agua quede debajo de lo permitido (20% en el sensor ultrasónico).

**Tabla 13**

*Funcionamiento del bebedero cuando el agua está limpia*

	Tiempo				
	4:06 PM	4:07 PM	4:09 PM	4:11 PM	4:13 PM
Sensores	Minuto 6	Minuto 8	Minuto 10	Minuto 12	Minuto 14
Sensor de turbidez	1.0	1.0	1.0	1.0	1.0
Sensor de nivel de líquido horizontal superior	Boya en la parte superior	Boya en la parte inferior	Boya en la parte inferior	Boya en la parte inferior	Boya en la parte inferior
Sensor de nivel de líquido horizontal inferior	Boya en la parte superior	Boya en la parte superior	Boya en la parte superior	Boya en la parte superior	Boya en la parte inferior
Ultrasónico	100%	80%	60%	40%	20%
Válvula solenoide superior	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula abierta
Válvula solenoide inferior	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula cerrada

**Nota.** Esta tabla muestra cómo funciona el bebedero cuando el agua está limpia.



### 3.5.1.3. Cuando el agua está turbia

El bebedero está lleno al cien por ciento, simulamos el proceso de consumo de agua del ganado que normalmente deja restos de comida que trae en su boca e incluso restos de excremento que los animalitos dejan en el bebedero. En tal sentido echaremos al agua pedacitos de hierba y líquido oscuro que permitan activar al sensor de turbidez.

**Tabla 14**

*Funcionamiento del bebedero cuando el agua está turbia*

	Tiempo			
	7:00 PM	7:01 PM	7:02 PM	7:03 PM
Sensores	Minuto 0	Minuto 1	Minuto 2	Minuto 3
Sensor de turbidez	1.0	2.0	3.5	5.0
Sensor de nivel de líquido horizontal superior	Boya en la parte superior	Boya en la parte inferior	Boya en la parte inferior	Boya en la parte inferior
Sensor de nivel de líquido horizontal inferior	Boya en la parte superior	Boya en la parte superior	Boya en la parte superior	Boya en la parte superior
Ultrasónico	100%	80%	60%	50%
Válvula solenoide superior	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula cerrada
Válvula solenoide inferior	Válvula cerrada	Válvula cerrada	Válvula cerrada	Válvula cerrada

**Nota.** Esta tabla muestra cómo funciona el bebedero cuando el agua está turbia.

Cuando la turbidez este en su máximo nivel, se enviará un correo al ganadero, para que haga la limpieza respectiva, y active el botón de desagüe.

#### 3.5.1.4. Vaciado de bebedero

El vaciado del bebedero tarda seis minutos estando lleno al cien por ciento, el vaciado del bebedero tiene que ser solicitado por el ganadero.

**Tabla 15**

*Funcionamiento del vaciado del bebedero*

Sensores	Tiempo		
	Minuto 1	Minuto 2	Minuto 3
Sensor de turbidez	3.5	3.5	1.0
Sensor de nivel de líquido horizontal superior	Boya en la parte inferior	Boya en la parte inferior	Boya en la parte inferior
Sensor de nivel de líquido horizontal inferior	Boya en la parte superior	Boya en la parte inferior	Boya en la parte inferior
Ultrasónico	33%	16%	0%
Válvula solenoide superior	Válvula cerrada	Válvula cerrada	Válvula cerrada
Válvula solenoide inferior	Válvula abierta	Válvula abierta	Válvula abierta

**Nota.** Esta tabla muestra cómo funciona el vaciado del bebedero.

### 3.5.2. Entrega

Encuesta al ganadero sobre los resultados del prototipo del bebedero automático.

**Tabla 16**

*Evaluación del funcionamiento del sistema en tiempo real*

FUNCIONALIDAD	INDICADORES			
	Muy Deficiente	Deficiente	Eficiente	Muy Eficiente
El sistema permite tener los bebedero llenos las 24 horas del día				
El sistema el tiempo real le permite llenar el bebedero de forma remota				
El sistema realiza el llenado de forma automática del bebedero				
El sistema le permite ahorrar tiempo en el llenado de bebederos				
El sistema produce derrames cuando realiza el llenado del bebedero				
El sistema en tiempo real le avisa cuando el agua está turbia				
El sistema permite visualizar el nivel de agua actual en cualquier momento				
El vaciado del bebedero lo puede realizar de forma remota				

**Nota.** Esta tabla muestra la encuesta que se aplicó al ganadero para evaluar el funcionamiento del sistema en tiempo real.

El ganadero marcó como “Muy eficiente” a los cuatro indicadores propuestos en la encuesta.

#### **Capítulo IV. Resultados y Discusión.**

Los resultados obtenidos son favorables, ya que gracias al uso de los sensores en conjunto se logró realizar el prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para controlar Nivel de Agua en la Quinta Fernández, el cual nos permite el ahorro de tiempo para el ganadero, ya que podrá controlar el vaciado o llenado del agua de forma remota accediendo a la aplicación Blynk que está instalada en su celular.

Así mismo se evitará desperdicios de agua, ya que al llegar al nivel máximo el prototipo de bebedero cierra automáticamente la válvula solenoide de llenado, y cuando llega al nivel mínimo, abre automáticamente la válvula de llenado, lo que permitirá que siempre haya agua disponible en el bebedero.

Por otro lado, cuando detecte que el agua esta turbia mandará automáticamente un correo al ganadero indicando el estado del agua y que debe ser cambiada. El ganadero tendrá dos opciones: ingresar a la aplicación Blynk y dar clic en el botón “desaguar” o presionar el botón físico que se encuentra a un lado del prototipo y comenzar el vaciado del agua.

En la presente investigación se ha propuesto un prototipo para el control del agua en un bebedero de ganado vacuno, y la importancia de visualizar el estado del nivel del agua en tiempo real desde una aplicación, reafirma lo expuesto por Osuna & Calderone (2018) quienes proponen un dispositivo controlado por una placa Arduino, la cual permite el monitoreo de la planta de tratamiento de agua potable, a través de internet, haciendo uso de un servidor en la nube, donde se puede verificar en tiempo real el estado actual del nivel del agua, pero no hace uso de un sensor de turbidez que permitiría conocer la calidad del agua, el cual es recomendable porque se está trabajando con agua potable. La misma recomendación iría para la investigación de Córdoba et al. (2020), que, si bien usa un sensor de flujo que permite obtener los valores del recorrido del agua que fluye por las tuberías en litros por minutos, también se debería saber la turbidez del agua ya que se habla del flujo de agua potable y esa agua será bebida por seres humanos debiendo estar en óptimas condiciones.

A diferencia de las investigaciones anteriores cuyo objetivo es medir el flujo del agua a través de un sistema, en nuestra investigación trata de controlar el nivel del agua en un bebedero de ganado vacuno, tratando no solo de medir la cantidad de agua sino sobre todo automatizando los procesos

de llenado y vaciado del bebedero, lo cual asemeja a la finalidad planteada en la investigación de Jara Neyra et al. (2019), quienes con el uso de un relé y una válvula solenoide admiten el fluido del agua manejando su apertura y su cierre, con la finalidad de evitar su derroche y controlar el consumo de agua dentro de un hogar, sin embargo la tecnología GSM utilizada puede ser de mucha dificultad para el usuario final, por eso es que se recomendaría realizar el control remoto mediante la aplicación Blynk, que permite visualizar el estado del nivel y la turbidez del agua, así como el control del llenado y vaciado automático.

Continuando con lo mencionado en los párrafos anteriores, sobre la importancia del control del agua, visualización del estado del nivel y la turbidez del agua de manera remota a través de una aplicación, Quezada Diaz & Salazar Santamaria (2018) quienes proponen en su proyecto un diseño de sistema de presión constante, el cual se encargará de llenar tanques de agua para edificios, es por ello que se le recomienda el uso del sensor de turbidez y el uso de algún aplicativo que permita el control de manera remota.

## Capítulo V. Conclusiones

Se desarrolló un prototipo de sistema en tiempo real de bebederos de ganado vacuno para controlar el nivel de agua haciendo uso la metodología Evolutiva por Prototipos, para el desarrollo del prototipo. De tal manera que se pudo construir tanto el hardware y el codificar el software en el lenguaje de programación Arduino.

Para diagnosticar la problemática del control del nivel del agua en la quinta Fernández se aplicó una entrevista al ganadero encargado, donde se determinó que el bebedero muchas veces se queda sin agua o en el otro extremo rebalsa, produciendo una inundación del corral y desperdicio del agua, generando más gastos. Tanto por la falta de agua o por la inundación del corral, trae como consecuencia una baja producción de leche o que el ganado se enferme.

El diseño del prototipo del sistema en tiempo real se desarrolló mediante el programa de diseño electrónico Fritzing, en donde se incluyó la placa ESP32, dos válvulas solenoides, dos sensores de nivel de líquido, un sensor de turbidez de agua, un sensor ultrasónico y dos pulsadores de parada. La placa ESP32 estará conectada a todos los demás circuitos electrónicos. El prototipo está compuesto por dos válvulas solenoides, una para el ingreso y otra para el desfogue del agua, cada válvula se conecta a un relé para determinar la apertura o cierre de las válvulas, también utiliza dos sensores de nivel de líquido, los cuales, gracias a la programación nos ayudarán a activar y desactivar el llenado del bebedero, para completar el funcionamiento del software se usa un sensor de turbidez, el cual nos permite conocer el nivel de turbidez en que se encuentra el agua del bebedero, también tenemos el sensor ultrasónico que estará ubicado en la parte superior del bebedero, que permitirá calcular el nivel actual del agua. Finalmente se ha usado dos pulsadores de parada, uno para detener todo el sistema y el otro para desaguar cuando sea necesario hacerlo de manera manual.

Para codificar el sistema en tiempo real, bajo el lenguaje de programación Arduino se declararon tres variables iniciales, las cuales corresponden al nivel máximo de agua, nivel mínimo de agua y valor de turbidez aceptables, posteriormente se desarrolló seis funciones, la función `sendWaterLevel` la que nos permitirá enviar el nivel de agua actual al servidor de Blynk, la función `measureWaterLevel` para medir el nivel de agua del Bebedero, función `sendTurbidity`, nos permitirá enviar el estado de la turbidez, la función `measureTurbidity` para

medir el valor de la Turbidez del agua, función `sendValveState` nos permite enviar el estado de las válvulas (abierta o cerrada), tanto la inferior como la superior y por último, la función `valveControl`, la cual nos ayudará a controlar las válvulas de apertura y cierre del agua.

Se realizaron las pruebas correspondientes para verificar el óptimo funcionamiento del sistema en tiempo real, haciendo las pruebas de funcionamiento del llenado de bebedero, cuando el agua está limpia, cuando el agua está turbia y vaciado del bebedero, encontrando que el 100% de las veces que se realizó el llenado automático total del bebedero demoró 6 minutos (el tiempo puede variar dependiendo de la presión del agua), si el valor de la turbidez es mayor o igual a 3 la válvula superior (válvula de llenado) se desactiva y envía un correo al ganadero donde indica que se ha sobrepasado los niveles de turbidez y que necesita ser limpiado, para vaciar el bebedero se puede hacer de dos maneras una desde el celular entrando al aplicativo Blynk, haciendo clic en la opción desaguar del bebedero y la otra opción es presionando el botón de color verde que se encuentra en el prototipo, el vaciado del bebedero estando lleno al 100% tarda 9 minutos en desaguar.

## **Capítulo VI. Recomendaciones.**

1. Se recomienda que el mantenimiento del sistema se dé cada 6 meses, y se realice la sustitución de cualquier conexión, sensor o válvula en mal estado o defectuosa, cada que se requiera.
2. Este sistema de bebedero depende del ancho de banda y la velocidad del Internet, para su monitoreo, se recomienda que la conexión WIFI este cerca a la placa ESP32, y así la eficiencia será mayor.
3. Cabe recalcar que para llevar a cabo este propósito se tuvo que evaluar distintos microcontroladores del mercado, siendo escogido el módulo ESP32, gracias a que este dispositivo es el más económico y completo, no necesita ningún periférico o placa extra para lograr una conexión WIFI, sin embargo, se recomienda a futuros investigadores revisar las placas microcontroladoras vigentes en el mercado ya que se actualizan y mejoran cada año.



## Bibliografía

- Agencia Agraria de Noticias. (2021, July 22). *Congreso aprobó Ley de promoción de la cadena productiva ganadero-lechera*. Agraria.Pe. <https://agraria.pe/noticias/congreso-aprobo-ley-de-promocion-de-la-cadena-productiva-gan-24953>
- Albarrán Trujillo, S. E. (2016). *Teoría de Sistemas - Unidad III: Modelos de Procesos de Desarrollo*. <http://ri.uaemex.mx/bitstream/handle/20.500.11799/64957/secme-17394.pdf?sequence=1>
- Arduino.cc. (2020). *Software Arduino*. Arduino. <https://www.arduino.cc/en/software>
- Arduino.cl. (n.d.). *Software de Arduino*. Arduino. Retrieved March 3, 2022, from <https://arduino.cl/programacion/>
- Arias Gonzáles, J. L. (2020). *Técnicas e instrumentos de investigación científica* (ENFOQUES CONSULTING EIRL (ed.); 1st ed.). Enfoques Consulting EIRL. <https://repositorio.concytec.gob.pe/handle/20.500.12390/2238>
- Blynk. (n.d.). *Blynk IoT platform*. Retrieved June 5, 2022, from <https://blynk.io/>
- Borjas Martínez, B. J., López Gutiérrez, N. H., & Rodríguez Rugama, M. P. (2017). Sistema automatizado para el área de Recursos Humanos de la Institución FundeMujer. *Universidad Nacional Autónoma de Nicaragua*. <https://repositorio.unan.edu.ni/9719/1/19031.pdf>
- Chavarry Vega, E. S., & Acedo Sosa, J. E. (2020). *Evaluación Energética de un Sistema de Bombeo de Líquidos Automatizado con Arduino en la empresa ARPE EIRL* [Universidad Nacional de Trujillo]. [https://dspace.unitru.edu.pe/bitstream/handle/UNITRU/16795/TESIS\\_FINAL\\_IQ\\_2021\\_ACEDO\\_CHAVARRY %281%29\\_protegido.pdf?sequence=1&isAllowed=y](https://dspace.unitru.edu.pe/bitstream/handle/UNITRU/16795/TESIS_FINAL_IQ_2021_ACEDO_CHAVARRY_%281%29_protegido.pdf?sequence=1&isAllowed=y)
- Ciro Edgardo, R., & Alejandro, E. (2022). Análisis de la capacidad de la placa ESP32 para integrar sistemas IoT descentralizados. *Elektron*, 10. <https://doi.org/10.37537/rev.elektron.6.1.142.2022>
- COLPOS. (2016). *Diseño y construcción de bebederos pecuarios*. <https://es.slideshare.net/demetriofernandez313/diseo-y-construccin-de-bebederos-pecuarios>
- Córdoba, J., Predyd, L., Bernal, C., & Mitre, R. (2020). Prototipo de sistema, utilizando arduino y gsm, para el monitoreo del flujo de agua potable en las tuberías, del IDAAN en la provincia de Coclé. *Revista Científica Guacamaya*, 4, 42–52. <https://revistas.up.ac.pa/index.php/guacamaya/article/view/1311/1076>
- Departamento de Arquitectura y Tecnología de Computadores. (n.d.). *Sistemas en Tiempo Real*. In *Informática Industrial* (Vol. 3, pp. 2–4). Retrieved March 2, 2022, from <http://icaro.eii.us.es/descargas/Tema6-parte1.pdf>
- Depaz Sandon, W. U. A. (2018). *Prototipo usando Tecnología arduino para medición de nivel de agua en lagunas peligrosas del parque Nacional Huascarán*. 2018. [http://repositorio.unasam.edu.pe/bitstream/handle/UNASAM/3361/T033\\_71573562\\_T.pdf?sequence=1&isAllowed=y](http://repositorio.unasam.edu.pe/bitstream/handle/UNASAM/3361/T033_71573562_T.pdf?sequence=1&isAllowed=y)
- Diario Oficial El Peruano. (2021, July 21). *LEY N° 31294*. <https://busquedas.elperuano.pe/download/url/ley-de-promocion-de-la-cadena-productiva-ganadero-lechera-ley-n-31296-1974970-5>
- Fernández Cirelli, A., Schenone, N., Pérez Carrera, A., & Volpedo, A. (2017). Calidad de agua para la producción de especies animales tradicionales y no tradicionales en Argentina. *AUGM DOMUS: Revista Científica Del Comité de Medio Ambiente Del Grupo Montevideo*, 9(revistas.unlp.edu.ar).

<https://revistas.unlp.edu.ar/domus/issue/view/359>

- Guaranga Allauca, J. E. (2018). *Diseño de un Sistema para seguridad de una vivienda mediante pasarela activada por voz y video a desarrollarse en la empresa "Rio Solar SMART ENERGY" de la ciudad de Riobamba* [Escuela Superior Politécnica de Chimborazo].  
<http://dspace.espol.edu.ec/bitstream/123456789/9220/1/108T0262.pdf>
- Havrylenko, S. B., Moltoni, A. F., Clemares, N., Espindola, A., & Fiorini Correa, J. G. (2022). Desarrollo de un prototipo electrónico inalámbrico para medir el nivel de agua en cuencas del sistema agroalimentario argentino. *XXX Congreso Latinoamericano de Hidráulica. 7 Al 11 de Noviembre de 2022, Foz de Iguazú, Brasil*, 1032–1040.  
<http://repositorio.inta.gob.ar:80/handle/20.500.12123/13752>
- Herrera Guaítara, R. J. (2011). *Estrategias de calidad de servicio al cliente y su incidencia en el volumen de ventas de la "Comercializadora e importadora Grupo Canguro CIA.LTDA"* [Universidad Técnica de Ambato]. [https://repositorio.uta.edu.ec/bitstream/123456789/1021/1/484 Ing.pdf](https://repositorio.uta.edu.ec/bitstream/123456789/1021/1/484%20Ing.pdf)
- Huayta Sucasaca, J. C., & Suaña Humpire, E. W. (2017). *Diseño e Implementación de un Sistema Scada para el control de nivel de agua para uso domótico mediante redes industriales* [Universidad Nacional del Altiplano Puno].  
[http://repositorio.unap.edu.pe/bitstream/handle/UNAP/4185/Huayta\\_Sucasaca\\_Jose\\_Carlos\\_Suaña\\_Humpire\\_Elmer\\_Wilson.pdf?sequence=1&isAllowed=y](http://repositorio.unap.edu.pe/bitstream/handle/UNAP/4185/Huayta_Sucasaca_Jose_Carlos_Suaña_Humpire_Elmer_Wilson.pdf?sequence=1&isAllowed=y)
- Instituto del Bien Común. (2017, November 23). *Instalación de bebederos para ganado evita contaminación de fuentes de agua en Oxapampa* /. <https://ibcperu.org/instalacion-de-bebederos-para-ganado-evita-contaminacion-de-fuentes-de-agua-en-oxapampa/>
- Instituto Nicaragüense de Tecnología Agropecuaria (INTA), & Instituto Nacional Tecnológico (INATEC). (2010). *Programa Manejo Sanitario Eficiente del Ganado Bovino: Principales Enfermedades* (L. Urbina Abaunza, A. Ballina G. Bencomo, F. Romero, & E. Reyes (eds.)).  
<https://www.fao.org/3/as497s/as497s.pdf>
- Jara Neyra, K., Ordinola Poicón, E. M., & Siancas Panta, K. M. (2019). *Prototipo de sistema de monitoreo y control de consumo de agua potable aplicando tecnología GSM* [Universidad Nacional de Piura]. <https://repositorio.unp.edu.pe/handle/UNP/1859>
- Langer, O. (2021, December 6). *La Industria Láctea Israelí: 100 Años de Innovación*. Ganaderia.Com.  
<https://www.ganaderia.com/destacado/la-industria-lactea-israeli-100-anos-de-innovacion>
- López, P. (2020, October 1). *¿Qué es una fuente de alimentación y para qué sirve?* Geeknetic.Es.  
<https://www.geeknetic.es/Fuente-de-alimentacion/que-es-y-para-que-sirve>
- Montenegro Ortiz, G. F. (2016). *Diseño electrónico de control para robots usando tecnología Open Source, plataforma Arduino* [Universidad de San Carlos de Guatemala].  
[http://www.repositorio.usac.edu.gt/5448/1/Gabriel Fernando Montenegro Ortiz.pdf](http://www.repositorio.usac.edu.gt/5448/1/Gabriel%20Fernando%20Montenegro%20Ortiz.pdf)
- Muñoz Llamuca, F. G., & Torres Torres, V. S. (2017). *Diseño y construcción de inductores de baja potencia de fabricación nacional aplicada en válvulas solenoides*. [Escuela Superior Politécnica del Litoral]. <https://www.dspace.espol.edu.ec/retrieve/128703/D-CD88631.pdf>
- Naylamp Mechatronic. (n.d.). *Fuente de alimentación DC 5V 1A*. Naylampmechatronics.Com. Retrieved March 4, 2022, from <https://naylampmechatronics.com/fuentes-switching-ac-dc/830-fuente-de-alimentacion-dc-5v-1a.html>
- Naylamp Mechatronics. (n.d.-a). *Arduboard Nano CH340G mini-USB*. Naylampmechatronics.Com. Retrieved March 4, 2022, from <https://naylampmechatronics.com/ardusystem-tarjetas/88-arduboard->

nano-ch340g-mini-usb.html

- Naylamp Mechatronics. (n.d.-b). *Fuente de alimentación DC 12V 1A*. Naylampmechatronics.Com. Retrieved March 4, 2022, from <https://naylampmechatronics.com/fuentes-switching-ac-dc/831-fuente-de-alimentacion-dc-12v-1a.html>
- Naylamp Mechatronics. (n.d.-c). *Módulo Relay 1CH 5VDC*. Naylampmechatronics.Com. Retrieved March 4, 2022, from [https://naylampmechatronics.com/drivers/297-modulo-relay-1-canal-5vdc.html?search\\_query=rele+5v&results=261](https://naylampmechatronics.com/drivers/297-modulo-relay-1-canal-5vdc.html?search_query=rele+5v&results=261)
- Naylamp Mechatronics. (n.d.-d). *Pulsador de parada de emergencia 15mm*. Naylampmechatronics.Com. Retrieved April 7, 2022, from <https://naylampmechatronics.com/accesorios-y-prototipado/659-pulsador-de-parada-de-emergencia-15mm.html>
- Naylamp Mechatronics. (n.d.-e). *Pulsador de parada de emergencia 15mm*. Naylampmechatronics.Com. Retrieved April 8, 2022, from <https://naylampmechatronics.com/accesorios-y-prototipado/659-pulsador-de-parada-de-emergencia-15mm.html>
- Naylamp Mechatronics. (n.d.-f). *Sensor de nivel de líquido horizontal*. Naylampmechatronics.Com. Retrieved April 8, 2022, from <https://naylampmechatronics.com/sensores-liquido/84-sensor-de-nivel-de-liquido-horizontal.html>
- Naylamp Mechatronics. (n.d.-g). *Sensor de turbidez de agua*. Retrieved March 3, 2022, from <https://naylampmechatronics.com/sensores-liquido/535-sensor-de-turbiedad-de-agua.html>
- Naylamp Mechatronics. (n.d.-h). *Sensor Ultrasonido*. Retrieved January 25, 2023, from <https://naylampmechatronics.com/sensores-proximidad/10-sensor-ultrasonido-hc-sr04.html>
- Naylamp Mechatronics. (n.d.-i). *Sensor Ultrasonido JSN-SR04T*. Retrieved May 8, 2022, from <https://naylampmechatronics.com/sensores-proximidad/326-sensor-ultrasonido-jsn-sr04t.html>
- Naylamp Mechatronics. (n.d.-j). *Válvula solenoide 1/2" 12VDC*. Naylampmechatronics.Com. Retrieved March 4, 2022, from [https://naylampmechatronics.com/valvulas/314-valvula-solenoide-1p2-pulg-12vdc.html?search\\_query=valvula+solenoide+1%2F2&results=255](https://naylampmechatronics.com/valvulas/314-valvula-solenoide-1p2-pulg-12vdc.html?search_query=valvula+solenoide+1%2F2&results=255)
- NaylaNaylamp Mechatronics. (n.d.). *NodeMCU-32 30-pin ESP32 WiFi*. Retrieved May 18, 2022, from <https://naylampmechatronics.com/espessif-esp/384-nodemc.-32-30-pin-esp32-wifi.html>. M. (n.d.). *NodeMCU-32 30-pin ESP32 WiFi*. Retrieved May 18, 2022, from <https://naylampmechatronics.com/espessif-esp/384-nodemcu-32-30-pin-esp32-wifi.html>
- Ocola Gadea, R. A., & Ramos Mendoza, G. W. (2018). *Implementación de una Aplicación Móvil complementada con la tecnología de Sensores Multiparamétricos con Arduino para monitorear la calidad del agua potable en el distrito de Laredo, Trujillo* [Universidad Privada del Norte]. [https://repositorio.upn.edu.pe/bitstream/handle/11537/14917/Ocola Gadea Raúl Alfonso - Ramos Mendoza Gorky Walter\\_parcial.pdf?sequence=5&isAllowed=y](https://repositorio.upn.edu.pe/bitstream/handle/11537/14917/Ocola%20Gadea%20Ra%C3%BAl%20Alfonso%20-%20Ramos%20Mendoza%20Gorky%20Walter%20parcial.pdf?sequence=5&isAllowed=y)
- Oficina de Software y Hardware Libre. (2014, May 30). *Fritzing: un programa 'open source' para el diseño electrónico*. Universidad Miguel Hernández de Elche. <http://oshl.umh.es/2014/05/30/fritzing-un-programa-open-source-para-el-diseno-electronico/>
- OMEGA. (n.d.). *Sensor de nivel*. Retrieved March 2, 2022, from <https://es.omega.com/prodinfo/sondas-de-nivel-medicion.html>
- Ordóñez Contreras, T. Y., & Ponce Jácome, B. J. (2019). *Diseño y desarrollo de prototipo de un dispositivo de medición de la calidad del agua de pozos artesanales aplicado en el recinto El Cacique de la Parroquia Junquillal del Cantón Salitre* [Universidad de Guayaquil].

- <http://repositorio.ug.edu.ec/bitstream/redug/45013/1/B-CINT-PTG-N.458> Ordóñez Contreras Tatiana Yuly . Ponce Jácome Bryan Jesús.pdf
- Ortiz Cruz, M. (2015). *Guía de entrevista y de observación*. [https://prezi.com/ooatecj5\\_fgt/guia-de-entrevista-y-de-observacion/](https://prezi.com/ooatecj5_fgt/guia-de-entrevista-y-de-observacion/)
- Osuna, J., & Calderone, J. (2018). *Monitoreo Planta de Agua mediante Sistema Arduino* . [http://sedici.unlp.edu.ar/bitstream/handle/10915/71679/Documento\\_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y](http://sedici.unlp.edu.ar/bitstream/handle/10915/71679/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y)
- Oversluijs Rengifo, D. F., & Rodríguez Najar, C. D. (2019). *Análisis y diseño de un prototipo de sistema de inventario de equipos informáticos de la empresa Frontera Energy - 2019* [Universidad Científica del Perú]. [http://repositorio.ucp.edu.pe/bitstream/handle/UCP/796/TRABAJO\\_FINAL\\_DANIEL\\_OVERSJLUIS\\_RENGIFO.pdf?sequence=4&isAllowed=y](http://repositorio.ucp.edu.pe/bitstream/handle/UCP/796/TRABAJO_FINAL_DANIEL_OVERSJLUIS_RENGIFO.pdf?sequence=4&isAllowed=y)
- Patricia Criollo, M. G. (2021, April 5). *¿qué es una guía de observacion y ejemplo?* Aleph.Org.Mx. <https://aleph.org.mx/que-es-una-guia-de-observacion-y-ejemplo>
- Pomposo Yanes, L. (2015). *Análisis de necesidades y propuesta de evaluación en línea de la competencia oral en inglés en el mundo empresarial* [Universidad Nacional de Educación a distancia]. [http://e-spacio.uned.es/fez/eserv/tesisuned:Filologia-Mlourdespomposo/POMPOSO\\_YANES\\_Lourdes\\_Tesis.pdf](http://e-spacio.uned.es/fez/eserv/tesisuned:Filologia-Mlourdespomposo/POMPOSO_YANES_Lourdes_Tesis.pdf)
- Pressman, R. (n.d.). *Ingeniería del Software: Un enfoque práctico* (P. Roig Vázquez & M. T. Zapata Terrazas (eds.); séptima). The McGraw-Hill. Retrieved March 22, 2022, from <http://cotana.informatica.edu.bo/downloads/ld-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF>
- Quezada Diaz, L. F., & Salazar Santamaria, J. I. (2018). *Diseño de un Sistema de Control de presión de agua en función de la demanda para el laboratorio de Ingeniería Electrónica - UNPRG* [Universidad Nacional Pedro Ruiz Gallo]. <https://repositorio.unprg.edu.pe/bitstream/handle/20.500.12893/2339/BC-TES-TMP-1215.pdf?sequence=1&isAllowed=y>
- Quingatuña Tasigchana, E. X. (2020). *Implementación de un prototipo para extracción de agua de un pozo mediante la medición de caudal y nivel del agua utilizando la plataforma Arduino y aplicación móvil Android* [Escuela Politécnica Nacional]. [https://bibdigital.epn.edu.ec/bitstream/15000/21326/1/CD\\_10841.pdf](https://bibdigital.epn.edu.ec/bitstream/15000/21326/1/CD_10841.pdf)
- Revista ElectroIndustria. (2019). Medición de nivel de líquidos. *Revista ElectroIndustria*. <http://www.emb.cl/electroindustria/articulo.mvc?xid=3388&ni=medicion-de-nivel-de-liquidos>
- Robots Argentina. (2020, July 8). *Módulos de relé y Arduino: Domótica (1)*. Robots-Argentina.Com.Ar. <http://robots-argentina.com.ar/didactica/modulos-de-rele-y-arduino-domotica-1/>
- Rodríguez Jiménez, A., & Pérez Jacinto, A. O. (2017). *Métodos científicos de indagación y de construcción del conocimiento*. 179–200. <http://www.scielo.org.co/pdf/ean/n82/0120-8160-ean-82-00179.pdf>
- Schweers, R. J. (2002). *Descripción en VHDL de arquitecturas para implementar el algoritmo CORDIC* [Universidad Nacional de La Plata]. [http://sedici.unlp.edu.ar/bitstream/handle/10915/3835/Tesis\\_completa.pdf?sequence=15&isAllowed=y](http://sedici.unlp.edu.ar/bitstream/handle/10915/3835/Tesis_completa.pdf?sequence=15&isAllowed=y)
- SENASA. (n.d.). *Sanidad Animal*. Retrieved March 1, 2022, from [https://www.argentina.gob.ar/sites/default/files/consumo\\_de\\_agua\\_segura\\_y\\_salud.pdf](https://www.argentina.gob.ar/sites/default/files/consumo_de_agua_segura_y_salud.pdf)

Stephen J. Young. (1982). *Real Time Languages: Design and Development*.

Torres Stuar, E. S., & Quintero Chocue, W. (2019). *Diseño e implementación de un dispensador remoto de alimento y agua para perros en la plataforma Arduino* .  
[https://repository.usc.edu.co/bitstream/handle/20.500.12421/4077/DISPENSADOR REMOTO ALIMENTO.pdf?sequence=2&isAllowed=y](https://repository.usc.edu.co/bitstream/handle/20.500.12421/4077/DISPENSADOR_REMOTO_ALIMENTO.pdf?sequence=2&isAllowed=y)

## Anexos.

### Anexo N°1

#### Operacionalización de Variables del prototipo de Sistema en Tiempo Real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández

**Figura 85**

*Operacionalización de variables*

Variable	Definición	Indicadores	Instrumento/ fórmula	Escala de medición
<b>Prototipo de sistema en tiempo real de bebederos de ganado vacuno.</b>	Según Stephen J. Young, (1982), un sistema en tiempo real es cualquier acción de procesamiento de información o un sistema que responde a estímulos externos en un intervalo de tiempo determinado.	Salida del Sensor de nivel de líquido horizontal superior	Encendido/apagado	Valor binomial
		Salida del Sensor de nivel de líquido horizontal inferior	Encendido/apagado	Valor binomial
		Salida de la Válvula solenoide superior	Encendido/apagado	Valor binomial
		Salida de la Válvula solenoide inferior	Encendido/apagado	Valor binomial
<b>Nivel de agua en la quinta Fernández</b>	Para Havrylenko et al., (2022), el nivel de agua o altura del agua, hace referencia a la altura de la superficie del agua en relación a cierta referencia específica, que puede ser registrada y operada sistemáticamente por una herramienta llamada limnógrafo.	Salida del Sensor Ultrasónico	El nivel de agua 0% al 100%	Porcentaje
		Salida del Sensor de turbidez	[1.0 a 5.0]	Intervalo

**Nota.** Esta tabla muestra la operacionalización de variables.

## Anexo N°2

### Guía de observación para realizar prototipo de Sistema en Tiempo Real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández

Objetivo: identificar el rendimiento del sistema en tiempo real de bebederos de ganado vacuno.

**Figura 86**

*Guía de observación para realizar el prototipo de sistema en tiempo real*

Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para Controlar Nivel de Agua en la Quinta Fernández			
<b>Datos Informativos</b> Fecha: Lugar:			
<b>Datos de rendimiento</b>			
Ítem	Indicador	Tiempo i	Observación
1	Sensor de turbidez	Rango de valores [1.0 a 5.0]	*
2	Sensor de nivel de líquido horizontal superior	Valor binomial: encendido/apagado	*
3	Sensor de nivel de líquido horizontal inferior	Valor binomial: encendido/apagado	*
4	Ultrasónico	El nivel de agua se va a medir en porcentajes del 0% al 100%	*
5	Válvula solenoide superior	Valor binomial: encendido/apagado	*
6	Válvula solenoide inferior	Valor binomial: encendido/apagado	*

**Nota.** Esta tabla muestra la guía de observación para realizar el sistema en tiempo real.

\*Se llenará ese campo si es que hubiera alguna observación atípica.

**Anexo N°3****Guía de entrevista para realizar prototipo de Sistema en Tiempo Real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández**

1. ¿Hace cuánto tiempo se dedica usted a la crianza de ganado vacuno?

---

---

2. ¿Cree usted que mantener los bebederos llenos de agua es un factor clave para la crianza del ganado? ¿Porqué?

---

---

3. ¿Cómo usted controla el llenado de los bebederos? ¿Cada cuánto tiempo los llenan?

---

---

4. ¿Cómo usted controla la limpieza del bebedero? ¿Cada que tiempo les hace limpieza?

---

---

5. ¿Qué problemas ha tenido usted con respecto al control del agua en los bebederos del ganado vacuno?

---

---

6. ¿Con el sistema en Tiempo Real, cree usted que le ayudará en el control del nivel del agua? ¿De qué manera le ayudaría?

---

---



#### Anexo N°4

### **Respuestas de la guía de entrevista para realizar el prototipo de Sistema en Tiempo Real de bebederos de ganado vacuno para controlar nivel de agua en la quinta Fernández, realizada al señor Simeon Tarrillo Guevara**

**Link:**<https://drive.google.com/drive/folders/1v7d3ffKziPrkIvEuYcwWqkFndQ5mIqpc?usp=sharing>

1. ¿Hace cuánto tiempo se dedica usted a la crianza de ganado vacuno?

Me dedico a la crianza de ganado vacuno, más de 50 años.

2. ¿Cree usted que mantener los bebederos llenos de agua es un factor clave para la crianza del ganado? ¿Porqué?

Si es necesario, porque bajaría su producción si falta el agua

3. ¿Cómo usted controla el llenado de los bebederos? ¿Cada cuánto tiempo los llenan?

Revisándolos uno por uno todos los días, lo llenamos dos veces diario

4. ¿Cómo usted controla la limpieza del bebedero? ¿Cada que tiempo les hace limpieza?

Cuando se ve que está sucio, se realiza la limpieza cuando tiene basura o sino una vez a la semana, dependiendo si el agua está limpia.

5. ¿Qué problemas ha tenido usted con respecto al control del agua en los bebederos del ganado vacuno?

Que a veces los bebederos se quedan sin nada de agua y otras veces que se inunda de agua el corral.

6. ¿Con el sistema en Tiempo Real, cree usted que le ayudará en el control del nivel del agua? ¿De qué manera le ayudaría?

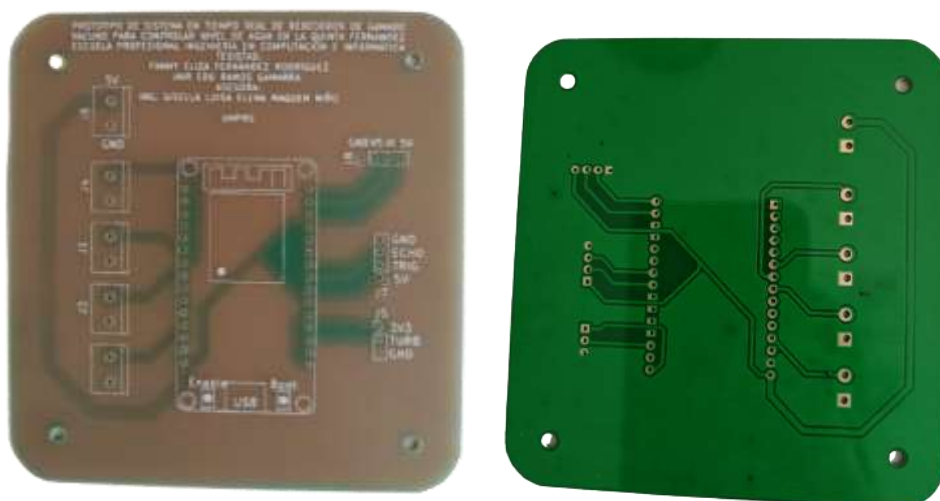
Si, por que ahorraría agua y tiempo, ya no estaría revisando de uno en uno

## Anexo N°5

### Imágenes del montaje del circuito del bebedero automático

**Figura 87**

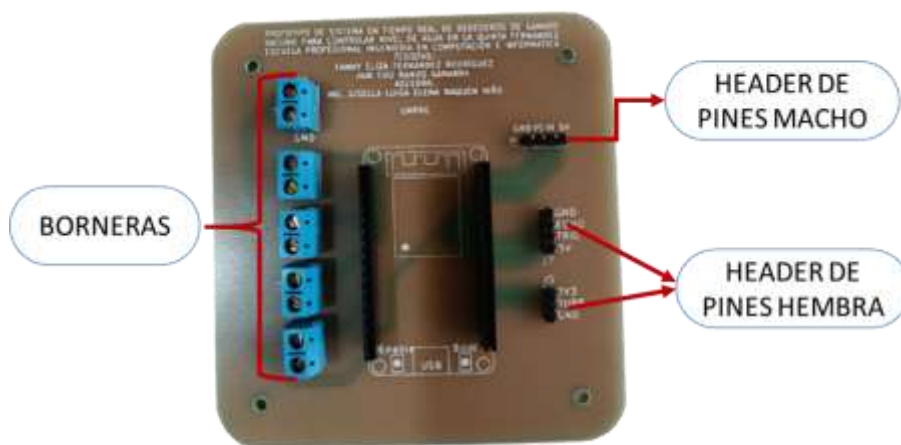
*Diseño de la placa*



**Nota.** Placa donde se soldará todo el circuito. Elaboración propia.

**Figura 88**

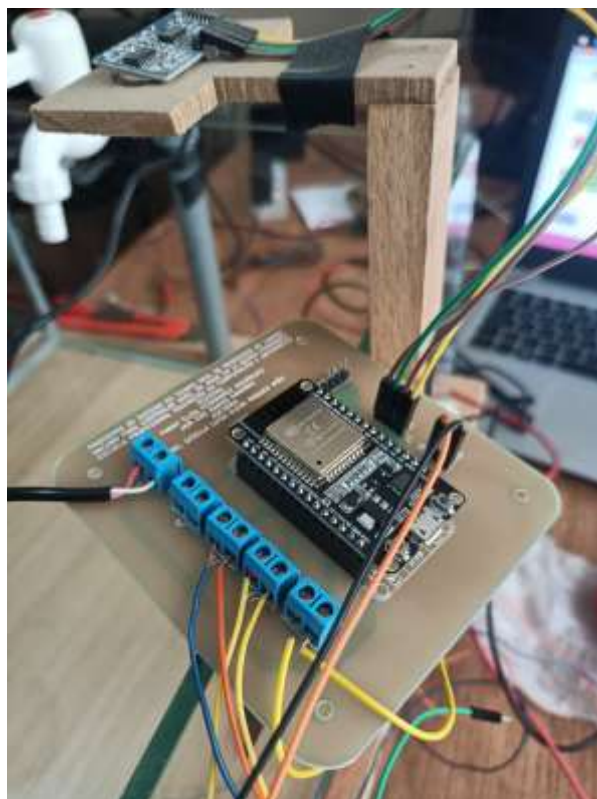
*Soldado de borneras y header de pines*



**Nota.** Soldado de borneras, header de pines hembras y machos. Elaboración propia.

**Figura 89**

*Sensores conectados a la placa*



**Nota.** Sensores conectados a la placa. Elaboración propia.

## Anexo N°6


**Respuestas de la encuesta para evaluar el correcto funcionamiento del proyecto de tesis “Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para controlar Nivel de Agua en la Quinta Fernández”, realizada al señor Simeon Tarrillo Guevara.**

**Figura 90**

*Respuesta de la encuesta para evaluar el correcto funcionamiento del proyecto*



**UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO**  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
ESCUELA PROFESIONAL DE INGENIERÍA EN COMPUTACIÓN E INFORMÁTICA




**ENCUESTA PARA EVALUAR EL CORRECTO FUNCIONAMIENTO DEL PROYECTO DE TESIS “PROTOTIPO DE SISTEMA EN TIEMPO REAL DE BEBEDEROS DE GANADO VACUNO PARA CONTROLAR NIVEL DE AGUA EN LA QUINTA FERNÁNDEZ”**

La presente encuesta tiene por objetivo conocer su opinión acerca del funcionamiento del sistema en tiempo real. Esta encuesta está dirigida al encargado del cuidado del ganado en la Quinta Fernández ubicada en la ciudad de Reque. Los resultados de esta investigación permitirán identificar la situación en la que se encuentra el sistema en tiempo real o si es necesario alguna mejora.

**Indicaciones:** Lea cuidadosamente cada pregunta y marque con un aspa (X) la respuesta que considere más adecuada.

FUNCIONALIDAD	INDICADORES			
	Muy Deficiente	Deficiente	Eficiente	Muy Eficiente
El sistema permite tener los bebedero llenos las 24 horas del día				X
El sistema el tiempo real le permite llenar el bebedero de forma remota				X
El sistema realiza el llenado de forma automática del bebedero				X
El sistema le permite ahorrar tiempo en el llenado de bebederos				X
El sistema evita derrames cuando llena el bebedero				X
El sistema en tiempo real le avisa cuando el agua está turbia				X
El sistema permite visualizar el nivel de agua actual en cualquier momento				X
El vaciado del bebedero lo puede realizar de forma remota				X

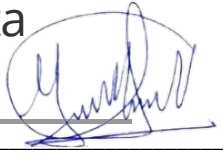
  
**Simeón Tarrillo Guevara**  
 DNI: 16594858  
**GANADERO**

  
**Fanny Eliza Fernández Rodríguez**  
 DNI: 75719188  
**TESISTA**

  
**Jair Edú Ramos Gamarra**  
 DNI: 73145554  
**TESISTA**

**Nota.** Encuesta realizada al ganadero, para saber el correcto funcionamiento del proyecto.  
Elaboración propia.

# Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno para Controlar Nivel de Agua en la Quinta Fernández



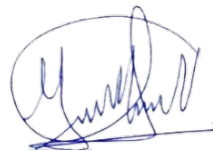
Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

## INFORME DE ORIGINALIDAD

17%	16%	3%	6%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

## FUENTES PRIMARIAS

1	<a href="https://hdl.handle.net">hdl.handle.net</a> Fuente de Internet	1%
2	<a href="https://1library.co">1library.co</a> Fuente de Internet	1%
3	<a href="https://bibdigital.epn.edu.ec">bibdigital.epn.edu.ec</a> Fuente de Internet	1%
4	<a href="https://sedici.unlp.edu.ar">sedici.unlp.edu.ar</a> Fuente de Internet	1%
5	Submitted to Universidad Nacional Pedro Ruiz Gallo Trabajo del estudiante	1%
6	<a href="https://repositoriosiidca.csuca.org">repositoriosiidca.csuca.org</a> Fuente de Internet	<1%
7	<a href="https://www.slideshare.net">www.slideshare.net</a> Fuente de Internet	<1%
8	<a href="https://www.coursehero.com">www.coursehero.com</a> Fuente de Internet	<1%



Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

<1 %

<1 %

<1 %

<1 %

<1 %

<1 %

<1 %

<1 %

<1 %

<1 %

<1 %

9

coggle.it

Fuente de Internet

10

repositorio.ug.edu.ec

Fuente de Internet

11

Submitted to Instituto Superior de Artes,  
Ciencias y Comunicación IACC

Trabajo del estudiante

12

docplayer.es

Fuente de Internet

13

repositorio.unasam.edu.pe

Fuente de Internet

14

repositorio.unan.edu.ni

Fuente de Internet

15

repositorio.unp.edu.pe

Fuente de Internet

16

www.dspace.unitru.edu.pe

Fuente de Internet

17

naylampmechatronics.com

Fuente de Internet

18

www.repositorio.usac.edu.gt

Fuente de Internet

19

pdfcookie.com

Fuente de Internet

20

Submitted to Universidad Cesar Vallejo

Trabajo del estudiante



&lt;1 %

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

&lt;1 %

21

[archive.org](https://archive.org)

Fuente de Internet

22

[repositorio.unsa.edu.pe](https://repositorio.unsa.edu.pe)

Fuente de Internet

&lt;1 %

23

[zaguan.unizar.es](https://zaguan.unizar.es)

Fuente de Internet

&lt;1 %

24

[dspace.esPOCH.edu.ec](https://dspace.esPOCH.edu.ec)

Fuente de Internet

&lt;1 %

25

[agraria.pe](https://agraria.pe)

Fuente de Internet

&lt;1 %

26

[repositorio.espe.edu.ec](https://repositorio.espe.edu.ec)

Fuente de Internet

&lt;1 %

27

Andrés Rodríguez Jiménez, Alipio Omar Pérez  
Jacinto. "Métodos científicos de indagación y  
de construcción del conocimiento", Revista  
EAN, 2017

Publicación

&lt;1 %

28

Submitted to Pontificia Universidad Católica  
del Ecuador - PUCE

Trabajo del estudiante

&lt;1 %

29

[repositorio.unprg.edu.pe](https://repositorio.unprg.edu.pe)

Fuente de Internet

&lt;1 %

Submitted to Universidad Carlos III de Madrid

30

Trabajo del estudiante



&lt;1 %

31

Submitted to Universidad Nacional del Centro del Peru

Trabajo del estudiante

Gisella Luisa Elena  
Maquen Niño  
DNI: 41747228  
ASESOR

&lt;1 %

32

oshl.umh.es

Fuente de Internet

&lt;1 %

33

repositorio.unprg.edu.pe:8080

Fuente de Internet

&lt;1 %

34

e-archivo.uc3m.es

Fuente de Internet

&lt;1 %

35

es.slideshare.net

Fuente de Internet

&lt;1 %

36

repositorio.udl.edu.pe

Fuente de Internet

&lt;1 %

37

repositorio.urp.edu.pe

Fuente de Internet

&lt;1 %

38

Submitted to AIEP

Trabajo del estudiante

&lt;1 %

39

"Applied Technologies", Springer Science and Business Media LLC, 2023

Publicación

&lt;1 %

40

dspace.utpl.edu.ec

Fuente de Internet

&lt;1 %





&lt;1 %

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

&lt;1 %

41

[www.dspace.espol.edu.ec](http://www.dspace.espol.edu.ec)

Fuente de Internet

42

[www.researchgate.net](http://www.researchgate.net)

Fuente de Internet

43

[repositorio.upec.edu.ec](http://repositorio.upec.edu.ec)

Fuente de Internet

&lt;1 %

44

[issuu.com](http://issuu.com)

Fuente de Internet

&lt;1 %

45

Submitted to Universidad Andina Nestor  
Caceres Velasquez

Trabajo del estudiante

&lt;1 %

46

[repositorio.ucp.edu.pe](http://repositorio.ucp.edu.pe)

Fuente de Internet

&lt;1 %

47

[repositorio.unap.edu.pe](http://repositorio.unap.edu.pe)

Fuente de Internet

&lt;1 %

48

[ibcperu.org](http://ibcperu.org)

Fuente de Internet

&lt;1 %

49

[repositorio.unae.edu.ec](http://repositorio.unae.edu.ec)

Fuente de Internet

&lt;1 %

50

[repositorio.uta.edu.ec](http://repositorio.uta.edu.ec)

Fuente de Internet

&lt;1 %

51

[www.mindmeister.com](http://www.mindmeister.com)

Fuente de Internet

&lt;1 %

52	Veronica Tapia, Raul Reinoso, Edison Carrillo, Dennis Cadas, Leonardo Criollo, Cristian Huilcatoma. "Translation system of voice and text to language of signs", 2017 12th Iberian Conference on Information Systems and Technologies (CISTI), 2017 Publicación	<1 %  Gisella Luisa Elena Maquen Niño DNI:41747228 ASESOR
53	<a href="http://repositorio.upn.edu.pe">repositorio.upn.edu.pe</a> Fuente de Internet	<1 %
54	<a href="http://www.pinterest.es">www.pinterest.es</a> Fuente de Internet	<1 %
55	Submitted to Eton School Trabajo del estudiante	<1 %
56	Submitted to Universidad Anahuac México Sur Trabajo del estudiante	<1 %
57	<a href="http://aquatropica.de">aquatropica.de</a> Fuente de Internet	<1 %
58	<a href="http://mafiadoc.com">mafiadoc.com</a> Fuente de Internet	<1 %
59	<a href="http://prezi.com">prezi.com</a> Fuente de Internet	<1 %
60	<a href="http://repositorio.unesum.edu.ec">repositorio.unesum.edu.ec</a> Fuente de Internet	<1 %
61	<a href="http://www.arnjournals.org">www.arnjournals.org</a> Fuente de Internet	



&lt;1 %

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

&lt;1 %

62

[lareferencia.info](http://lareferencia.info)

Fuente de Internet

63

[Submitted to Universidad Nacional de San  
Cristóbal de Huamanga](#)

Trabajo del estudiante

&lt;1 %

64

[Submitted to Universidad San Ignacio de  
Loyola](#)

Trabajo del estudiante

&lt;1 %

65

[Submitted to BENEMERITA UNIVERSIDAD  
AUTONOMA DE PUEBLA BIBLIOTECA](#)

Trabajo del estudiante

&lt;1 %

66

[Submitted to Universidad Continental](#)

Trabajo del estudiante

&lt;1 %

67

[Submitted to Universidad Estatal a Distancia](#)

Trabajo del estudiante

&lt;1 %

68

[www3.uah.es](http://www3.uah.es)

Fuente de Internet

&lt;1 %

69

[Submitted to Ministerio de Educación de Perú  
- COAR](#)

Trabajo del estudiante

&lt;1 %

70

[qastack.mx](http://qastack.mx)

Fuente de Internet

&lt;1 %

71

[renati.sunedu.gob.pe](http://renati.sunedu.gob.pe)

Fuente de Internet



&lt;1 %

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

&lt;1 %

72

[tesis.ucsm.edu.pe](https://tesis.ucsm.edu.pe)

Fuente de Internet

73

[jultika.oulu.fi](https://jultika.oulu.fi)

Fuente de Internet

&lt;1 %

74

[www.march.es](https://www.march.es)

Fuente de Internet

&lt;1 %

75

Ugarte Diaz, Diego Paul. "Desarrollo de un controlador inteligente para un bastidor de osmosis inversa de una planta desalinizadora de agua de mar.", Pontificia Universidad Catolica del Peru - CENTRUM Catolica (Peru), 2021

Publicación

&lt;1 %

76

[doaj.org](https://doaj.org)

Fuente de Internet

&lt;1 %

77

[ferretronica.com](https://ferretronica.com)

Fuente de Internet

&lt;1 %

78

[repositorio.uncp.edu.pe](https://repositorio.uncp.edu.pe)

Fuente de Internet

&lt;1 %

79

[s3platform.jrc.ec.europa.eu](https://s3platform.jrc.ec.europa.eu)

Fuente de Internet

&lt;1 %

80

[www.eluniverso.com](https://www.eluniverso.com)

Fuente de Internet

&lt;1 %

81 [www.indecopi.gob.pe](http://www.indecopi.gob.pe)

Fuente de Internet



<1 %

82 [xfactortech.com](http://xfactortech.com)

Fuente de Internet

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

<1 %

83 Monroy Bañuelos Lizeth. "Diseño e implementación de un sistema de medición, presentación y almacenamiento de información ambiental en tiempo real", TESIUNAM, 2015

Publicación

<1 %

84 Sánchez, Janet Rocío Pajuelo | Parado, Sandra Ponce | Suazo, Karen Raquel Reynoso. "Percepción Visual y Competencia Matemática en Niños Preescolares de dos Instituciones Públicas en Villa el Salvador-2016", Pontificia Universidad Catolica del Peru - CENTRUM Catolica (Peru), 2022

Publicación

<1 %

85 Submitted to Universidad Europea de Madrid

Trabajo del estudiante

<1 %

86 [bibliotecadigital.udea.edu.co](http://bibliotecadigital.udea.edu.co)

Fuente de Internet

<1 %

87 [manualzz.com](http://manualzz.com)

Fuente de Internet

<1 %

88 [repositorio.uceva.edu.co](http://repositorio.uceva.edu.co)

Fuente de Internet

<1 %

89 repositorio.ucsg.edu.ec

Fuente de Internet



<1 %

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

90 repositorio.usmp.edu.pe

Fuente de Internet

<1 %

91 repositorio.utc.edu.ec

Fuente de Internet

<1 %

92 tesis.ipn.mx

Fuente de Internet

<1 %

93 www.fao.org

Fuente de Internet

<1 %

94 www.mangusta4.yoyo.pl

Fuente de Internet

<1 %

95 "Inter-American Yearbook on Human Rights /  
Anuario Interamericano de Derechos  
Humanos, Volume 12 (1996)", Brill, 1998

Publicación

<1 %

96 "Inter-American Yearbook on Human Rights /  
Anuario Interamericano de Derechos  
Humanos, Volume 26 (2010)", Brill, 2014

Publicación

<1 %

97 Mora, Mariana. "Decolonizing politics:  
Zapatista indigenous autonomy in an era of  
neoliberal governance and low intensity  
warfare", Proquest, 20111108

Publicación

<1 %



&lt;1 %

Gisella Luisa Elena  
Maquen Niño  
DNI:41747228  
ASESOR

&lt;1 %

98

[digibug.ugr.es](http://digibug.ugr.es)

Fuente de Internet

99

[lamenteesmaravillosa.com](http://lamenteesmaravillosa.com)

Fuente de Internet

100

[pceiberica.en.ecplaza.net](http://pceiberica.en.ecplaza.net)

Fuente de Internet

&lt;1 %

101

[repositorio.espe.edu.ec:8080](http://repositorio.espe.edu.ec:8080)

Fuente de Internet

&lt;1 %

102

[repositorio.sangregorio.edu.ec](http://repositorio.sangregorio.edu.ec)

Fuente de Internet

&lt;1 %

103

[repositorio.ucv.edu.pe](http://repositorio.ucv.edu.pe)

Fuente de Internet

&lt;1 %

104

[repositorio.usil.edu.pe](http://repositorio.usil.edu.pe)

Fuente de Internet

&lt;1 %

105

[repositorio.utn.edu.ec](http://repositorio.utn.edu.ec)

Fuente de Internet

&lt;1 %

106

[www.ceei.es](http://www.ceei.es)

Fuente de Internet

&lt;1 %

107

[www.hartig-helling.de](http://www.hartig-helling.de)

Fuente de Internet

&lt;1 %

108

[www.intermnet.ua.es](http://www.intermnet.ua.es)

Fuente de Internet

&lt;1 %

109

[www.mef.gob.pe](http://www.mef.gob.pe)

Fuente de Internet

&lt;1 %

110

[www.sdnf.undp.org](http://www.sdnf.undp.org)

Fuente de Internet



&lt;1 %

111

["Inter-American Yearbook on Human Rights /](#)[Anuario Interamericano de Derechos](#)[Humanos, Volume 27 \(2011\)", Brill, 2015](#)

Publicación

Gisella Luisa Elena

Maquen Niño

DNI: 41747228

ASESOR

&lt;1 %

112

Blanco Camarillo Juan Antonio, Pinales

Landeros Daniela. "Modelo funcional de un

sistema para el cuidado de una planta de

ornato : Careplant", TESIUNAM, 2014

Publicación

&lt;1 %

113

[qdoc.tips](http://qdoc.tips)

Fuente de Internet

&lt;1 %

Excluir citas

Activo

Excluir coincidencias &lt; 5 words

Excluir bibliografía

Activo





## Recibo digital


Este recibo confirma que su trabajo ha sido recibido por **Turnitin**. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Fernández Rodríguez Fanny - Ramos Gamarra Jair  
Título del ejercicio: PRE GRADO EPICI  
Título de la entrega: Prototipo de Sistema en Tiempo Real de Bebederos de Gana...  
Nombre del archivo: AL\_FERN\_NDEZ\_RODR\_GUEZ\_FANNY\_ELIZA\_-\_RAMOS\_GAMAR...  
Tamaño del archivo: 5.46M  
Total páginas: 124  
Total de palabras: 21,535  
Total de caracteres: 135,390  
Fecha de entrega: 19-abr.-2023 01:43p. m. (UTC-0500)  
Identificador de la entrega... 2069575336

1

UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
ESCUELA PROFESIONAL DE INGENIERÍA EN  
COMPUTACIÓN E INFORMÁTICA



TESIS

"Prototipo de Sistema en Tiempo Real de Bebederos de Ganado Vacuno  
para Controlar Nivel de Agua en la Quinta Fernández"

Presentado para optar el Título Profesional de:  
Ingeniero (a) en Computación e Informática

Investigadores:  
Fernández Rodríguez Fanny Eliza  
Ramos Gamarra Jair Edú

Asesor:  
Dra. Ing. Maquen Niño Gisela Luisa Elena

Lambayeque, 2022

**Dra. Gisella Luisa Elena Maquen Niño**  
**DNI 41747228**