

**UNIVERSIDAD NACIONAL MICAELA BASTIDAS DE APURÍMAC**  
**FACULTAD DE INGENIERÍA**

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA INFORMÁTICA Y SISTEMAS



Tesis

Automatización de riego tecnificado por goteo, con el desarrollo e implementación de un software con tecnología arduino, 2021

Presentado por:

Hidania Grimaldina Oscco Soto

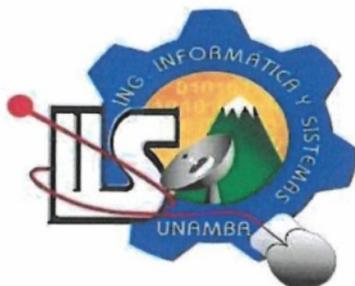
Para optar el título de Ingeniero Informático y Sistemas

Abancay, Perú

2025



UNIVERSIDAD NACIONAL MICAELA BASTIDAS DE APURÍMAC  
FACULTAD DE INGENIERÍA  
ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA INFORMÁTICA Y  
SISTEMAS



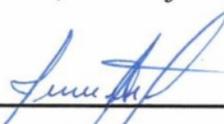
TESIS

**Automatización de riego tecnificado por goteo, con el desarrollo e implementación de un software con tecnología arduino, 2021**

Presentado por **Hidania Grimaldina Oscco Soto**, para optar el título de: Ingeniero Informático y Sistemas

Sustentado y aprobado el 04 de setiembre de 2024, ante el jurado evaluador:

**Presidente:**

  
\_\_\_\_\_  
*Mg. Eyelyn Naida Luque Ochoa*

**Primer miembro:**

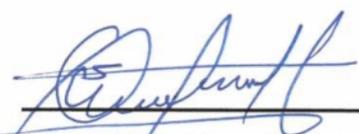
  
\_\_\_\_\_  
*Dr. Ronald Alberto Rentería Ayquipa*

**Segundo miembro:**

  
\_\_\_\_\_  
*Dr. Lintol Contreras Salas*

**Asesores:**

  
\_\_\_\_\_  
*Mag. Marleny Peralta Ascue*

  
\_\_\_\_\_  
*M.Sc. Yhon Fuentes Huamán*

---

## CONSTANCIA DE ORIGINALIDAD N° 151-2024

La Universidad Nacional Micaela Bastidas de Apurímac, a través de la Unidad de Investigación de la Facultad de Ingeniería declara que, la Tesis intitulada **AUTOMATIZACIÓN DE RIEGO TECNIFICADO POR GOTEJO, CON EL DESARROLLO E IMPLEMENTACIÓN DE UN SOFTWARE CON TECNOLOGÍA ARDUINO, 2021**, presentado por la Bach. **Hidania Grimaldina OSCCO SOTO**, Para optar el Título de **Ingeniero Informático y Sistemas**; ha sido sometido a un mecanismo de evaluación y verificación de similitud, a través del Software Turnitin, siendo el índice de similitud **ACEPTABLE de (10%)** por lo que, cumple con los criterios de originalidad establecidos por la Universidad.

Abancay, 29 de agosto del 2024



UNIVERSIDAD NACIONAL MICAELA BASTIDAS  
DE APURIMAC  
**Dr. Lintol Contreras Salas**  
DIRECTOR(E) DE LA UNIDAD DE INVESTIGACION  
FACULTAD DE INGENIERIA

C. c.  
Archivo  
REG. N° 596

## **Agradecimiento**

*Agradecer a Dios, porque él sabe cómo tener detalles conmigo y a mi madre por su paciencia, comprensión y amor reflejado en insistencia. A mi asesor y co-asesor por la guía, ánimo e impulso, a mis docentes por su contribución en mi formación profesional y a mí misma, por hacer posible este logro y, en general, a todos los que contribuyeron en la realización de esta investigación.*



## **Dedicatoria**

*Dedicado a mi madre por su interminable paciencia y comprensión, a mis hermanos, a mi pareja e hija, a mi familia en general porque me brindaron su apoyo incondicional, por compartir sus buenos y malos momentos y cada una de las personas que contribuyeron en mi crecimiento como personas y profesional.*



Automatización de riego tecnificado por goteo, con el desarrollo e implementación de  
unsoftware con tecnología Arduino, 2021

Línea de Investigación: Ingeniería de software e innovación tecnológica

Esta publicación está bajo una Licencia Creative Commons



## ÍNDICE

	<b>Pág.</b>
<b>INTRODUCCIÓN</b>	1
<b>RESUMEN</b>	3
<b>ABSTRACT</b>	5
<b>CAPÍTULO I</b>	7
<b>PLANTEAMIENTO DEL PROBLEMA</b>	7
1.1 Descripción del problema	7
1.2 Enunciado del problema	8
1.2.1 Problema general	8
1.2.2 Problemas específicos	8
1.3 Justificación de la investigación	9
<b>CAPÍTULO II</b>	10
<b>OBJETIVOS E HIPÓTESIS</b>	10
2.1 Objetivos de la investigación	10
2.1.1 Objetivo general	10
2.1.2 Objetivo específicos	10
2.2 Hipótesis de la investigación	10
2.2.1 Hipótesis general	10
2.2.2 Hipótesis específicas	10
2.3 Operacionalización de variables	11
<b>CAPÍTULO III</b>	12
<b>MARCO TEÓRICO REFERENCIAL</b>	12
3.1 Antecedentes	12
3.1.1 Antecedentes internacionales	12
3.1.2 Antecedentes nacionales	14
3.1.3 Antecedente local	17
3.2 Marco teórico	18
3.2.1 Riego por goteo	18
3.2.2 Sistema de riego	19
3.2.3 Eficiencia de riego	20
3.2.4 Arduino	22
	I



3.2.4.1	Microcontrolador Arduino-ATmega 328	22
3.2.4.2	Memorias del arduino	24
3.2.5	Protocolos de comunicación	25
3.2.6	Alimentación	26
3.2.7	Componentes del arduino	27
3.2.8	Tecnología de identificación por radiofrecuencia RFID	29
3.2.9	Eficiencia de desempeño del software	29
3.3	Marco conceptual	31
3.3.1	Actuador	31
3.3.2	Evotranspiración	32
3.3.3	Diseño agronómico	32
3.3.4	Diseño hidráulico	32
3.3.5	Relación agua-suelo-planta	32
3.3.6	Placas de prototipado	32
3.3.7	Sensor	33
<b>CAPÍTULO IV</b>		34
<b>METODOLOGÍA</b>		34
4.1	Tipo y nivel de investigación	34
4.1.1	Tipo de investigación	34
4.1.2	Nivel de estudio	34
4.2	Diseño de la investigación	34
4.3	Población y muestra	35
4.3.1	Población	35
4.3.2	Muestra	35
4.3.3	Técnica	36
4.3.4	Instrumentos	36
<b>CAPÍTULO V</b>		38
<b>RESULTADOS Y DISCUSIONES</b>		38
5.1	Análisis de resultados	38
5.1.1	Diseño agronómico para determinar la cantidad de agua para el cultivo	38
5.1.2	Diseño hidráulico para el riego tecnificado por goteo	48
5.1.3	Programación en arduino UNO para el funcionamiento del riego por goteo	51
5.1.4	Interfaz del software con tecnología arduino	54
5.1.5	Resultado del diseño hidráulico	55
5.1.6	Análisis de resultados de la investigación de la calidad de la eficiencia de desempeño del software con tecnología arduino.	56
5.2	Contrastación de hipótesis	58
5.2.1	Contrastación hipótesis para la eficiencia de riego	58
		II



5.2.2	Contrastación hipótesis para el tiempo de riego	58
5.2.3	Contrastación de hipótesis para el consumo de agua	62
5.2.4	Contrastación de hipótesis para el nivel de humedad	66
5.3	Discusión	69
	<b>CAPÍTULO VI</b>	73
	<b>CONCLUSIONES Y RECOMENDACIONES</b>	73
6.1	Conclusiones	73
6.2	Recomendaciones	74
	<b>REFERENCIAS BIBLIOGRÁFICAS</b>	75
	<b>ANEXOS</b>	78



## ÍNDICE DE TABLAS

	<b>Pág.</b>
Tabla 1 — Cuadro de operacionalización de variables	11
Tabla 2 — Resultado de medir la eficiencia de desempeño del software con tecnología arduino	57
Tabla 3 — Datos estadísticos	59
Tabla 4 — Datos estadísticos con sistema y sin sistema	62
Tabla 5 — Datos estadísticos con sistema y sin sistema	66

## ÍNDICE DE FIGURAS

	<b>Pág.</b>
Figura 1— Tipos de software de riegos	19
Figura 2 — Placa de arduino uno	24
Figura 3 — Componentes del Arduino	29
Figura 4 — Placa de prototipado (protoboards)	33
Figura 5 — Prueba de hipótesis	37
Figura 6 — Ubicación del lugar donde se exportará los datos de precipitaciones y clima	39
Figura 7 — Ubicación Abancay	39
Figura 8 — Exportación de datos de clima y precipitación	40
Figura 9 — Datos de la estación de ABANCAY	40
Figura 10 — Método FAO penman monteith	41
Figura 11 — Importación de precipitación	42
Figura 12 — Histograma de las precipitaciones de la ciudad de Abancay	42
Figura 13 — Documento de la FAO para el cálculo de la evapotranspiración	43
Figura 14 — Duración de las etapas de crecimiento del cultivo	44
Figura 15 — Diseño de cultivo remolacha	44
Figura 16 — Requerimiento de agua para el cultivo de remolacha	45
Figura 17—Requerimiento de agua para el cultivo de remolacha	45
Figura 18 — Interfaz de programación del cultivo	46
Figura 19 — Requerimiento de agua para el cultivo, y la ET0	47
Figura 20 — Programación de cultivo	47
Figura 21 — Diseño hidráulico de riego por goteo	49
Figura 22 — Diseño de válvula	50
Figura 23 — Diseño de DHT	50
Figura 24 — Esquema sensor de humedad	51
Figura 25 — Interfaz gráfica de usuario del software con tecnología arduino	54
Figura 26 — Ramal a ramal 40cm de un gotero a otro	55
Figura 27 — Líneas terciarias instaladas con la tecnología de riego por goteo	55
Figura 28 — Goteros en cada subunidad	56
Figura 29 — Sensor de humedad instalado en la raíz de una remolacha	56
	V



Figura 30 — Curva de región crítica de la hipótesis general	61
Figura 31 — Gráfico de tiempo de riego con la aplicación y sin la aplicación	61
Figura 32 — Gráfico de región crítica	64
Figura 33 — Gráfico de pérdida de agua con y sin el sistema	65
Figura 34 — Gráfico de región crítica o aceptación	68
Figura 35 — Gráfico de dispersión de porcentaje de humedad con y sin el software	68
Figura 36 — Regresión lineal del tiempo de llegadas del agua a los vegetales	70
Figura 37 — Regresión lineal de la cantidad de agua con y sin la aplicación	71
Figura 38 — Gráfico del control de humedad del suelo con la aplicación	72
Figura 39 — Terreno	78
Figura 40 — Preparación de terreno	78
Figura 41 — Sembrado de remolacha	79
Figura 42 — Diseño de las laterales y las subunidades	80
Figura 43 — Materiales para la instalación del riego	80
Figura 44 — Goteros empalmes y tubería HDP de ½	81
Figura 45 — Surcos elaborados de acuerdo al diseño agronómico e hidráulico.	82
Figura 46 — Distribución de las subunidades del sembrío	82
Figura 47 — Pruebas del riego por goteo con las subunidades	83
Figura 48 — Distribución de los goteros de acuerdo al manual de la FAO	84
Figura 49 — Distribución de los goteros en remolachas	84



## INTRODUCCIÓN

La técnica de riego por goteo es altamente beneficiosa para las empresas agroexportadoras y los pequeños agricultores. La eficiencia en el uso del agua en los sistemas de riego representa uno de los desafíos más significativos a nivel mundial en la actualidad. Para abordar este problema, es fundamental implementar técnicas que utilicen adecuadamente la programación del riego, con el fin de mejorar la eficiencia en el uso del agua, conservándola cuando sea necesario y reducir al mínimo su consumo excesivo. Es en ese contexto, que la presente investigación se centra en la automatización del riego tecnificado por goteo a través del desarrollo e implementación de un software basado en la tecnología Arduino.

Durante la ejecución del proyecto, se abordaron dos aspectos fundamentales: el diseño hidráulico y el diseño agronómico. En el diseño hidráulico, se determinó la disposición de las tuberías, definiendo sus características y medidas, así como la selección de la bomba de agua más adecuada en función a sus capacidades. Asimismo, se utilizaron sensores de humedad y temperatura para monitorear las condiciones del suelo.

Por otro lado, en el diseño agronómico se establecieron los parámetros necesarios para la implementación del riego tecnificado por goteo. Se trazó la ruta del agua desde la captación hasta las plantas, definiendo un sistema eficiente de distribución del agua y se evaluó cuanta capacidad de agua requiere el tipo de sembrío. De este modo, se desarrolló e implementó el software basado en la tecnología Arduino, logrando así mejorar significativamente la eficiencia de riego para cultivos.

La investigación consta de seis capítulos: el primer capítulo se presenta la descripción del problema, el enunciado del problema y la justificación de la investigación.

El segundo capítulo se presenta los objetivos, las hipótesis y la operacionalización de las variables.

En el tercer capítulo se presenta el marco teórico de la investigación, el cual incluye antecedentes internacionales, nacionales y locales que se utilizan como referencia y contraste con este estudio. Además, se abordan los términos más relevantes de la investigación definidas en el marco conceptual.



En el cuarto capítulo se presenta la metodología de la investigación.

En el quinto capítulo se presentan los resultados y discusión.

En el sexto capítulo se presentan las conclusiones y recomendaciones.



## RESUMEN

La investigación se realizó con el objetivo de automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, con la finalidad de mejorar la eficiencia del riego. En este proyecto se desarrolló un sistema de riego tecnificado por goteo utilizando Arduino como plataforma de programación para controlar las válvulas de agua. Estas válvulas desempeñan un papel importante en el diseño hidráulico, al regular el flujo de agua de manera precisa. Además, se integraron sensores de humedad y temperatura para controlar las condiciones del suelo, cantidad y duración del riego en función de las necesidades de las plantas. Estos datos permitieron abrir o cerrar las válvulas de agua de manera automatizada. Para determinar los requerimientos hídricos del cultivo, se realizó un análisis agronómico del tipo de siembra, en este caso, de betarraga (también conocida como remolacha). Se utilizaron herramientas como el programa CLIMWAT y CROPWAT, junto con los estándares de la FAO, para calcular las necesidades de agua del cultivo. El diseño hidráulico se llevó a cabo en un área aproximada de 7m de ancho por 11m de largo, lo que resultó en un área total de 77m<sup>2</sup>. Se instalaron 19 goteros en cada fila, con un total de 11 filas y un espaciado de 50 cm entre ellas. Los resultados mostraron una mejora en la eficiencia del riego tecnificado por goteo. Se observó una desviación estándar de 16.15s en el sistema de riego automatizado, en comparación con 14.69s sin el sistema. Además, los tiempos de llegada al primer y último gotero fueron de 35.0s y 81.0s, respectivamente. En cuanto al uso adecuado del agua, se encontró una correlación de  $r=0.208$  entre el control de agua con el sistema y sin el sistema, lo que indica que existe una mejora en el control. La media de la cantidad de agua utilizada fue de 0.3800m<sup>3</sup> con la aplicación del riego tecnificado, mientras que sin él fue de 0.65m<sup>3</sup>, lo que representa una diferencia de 0.27m<sup>3</sup> por segundo.

En relación al control de la humedad del suelo, se observó que, con el sistema de riego tecnificado, se obtuvo un promedio de humedad del suelo de 27.0, mientras que, sin el sistema, el promedio fue de 11.6. Esto demuestra que el control y manejo preciso de la cantidad de agua aplicada contribuye significativamente a regular la humedad del suelo de manera



eficiente.

**Palabras clave:** *sistema de riego automatizado, Arduino, riego tecnificado por goteo, diseño agronómico, diseño hidráulico.*



## ABSTRACT

The research was carried out with the objective of automating drip irrigation through the development and implementation of software with Arduino technology, with the aim of improving irrigation efficiency. In this project, a technical drip irrigation system was developed using Arduino as a programming platform to control the water valves. These valves play an important role in hydraulic design by precisely regulating water flow. In addition, humidity and temperature sensors were integrated to control soil conditions, quantity and duration of irrigation depending on the needs of the plants. This data allowed the water valves to be opened or closed automatically. To determine the water requirements of the crop, an agronomic analysis of the type of planting was carried out, in this case, beet (also known as beet). Tools such as the CLIMWAT and CROPWAT programs, along with FAO standards, were used to calculate the crop's water needs. The hydraulic design was carried out in an area of approximately 7m wide by 11m long, resulting in a total area of 77m<sup>2</sup>. 19 drippers were installed in each row, with a total of 11 rows and a spacing of 50 cm between them. The results showed an improvement in the efficiency of technical drip irrigation. A standard deviation of 16.15s was observed in the automated irrigation system, compared to 14.69s without the system. Furthermore, the arrival times to the first and last dropper were 35.0s and 81.0s, respectively. Regarding the proper use of water, a correlation of  $r=0.208$  was found between water control with the system and without the system, which indicates that there is an improvement in control. The average amount of water used was 0.3800m<sup>3</sup> with the application of technical irrigation, while without it was 0.65m<sup>3</sup>, which represents a difference of 0.27m<sup>3</sup> per second. In relation to the control of soil humidity in technical irrigation, it was observed that with the technical irrigation system, an average soil humidity of 27.0 was obtained, while, without the system, the average was 11.6. These data show a great difference, which is due to the precise control and management of the amount of water applied. This control contributes significantly to regulating soil moisture efficiently.



**Keywords:** *automated irrigation system, Arduino, technical drip irrigation, agronomic design, hydraulic design.*



## **CAPÍTULO I**

### **PLANTEAMIENTO DEL PROBLEMA**

#### **1.1 Descripción del problema**

En la actualidad, el riego tecnificado por goteo es una técnica ampliamente utilizada en la agricultura para suministrar agua de manera precisa a las plantas. Sin embargo, el proceso de controlar manualmente el riego puede ser laborioso y propenso a errores. Por consiguiente, surge la necesidad de automatizar este proceso con el fin de mejorar la eficiencia del riego tecnificado por goteo.

La ausencia de un sistema automatizado conlleva a dificultades en la programación y control constante, lo que podría resultar en un uso ineficiente del agua y en una distribución desigual de los recursos hídricos.

La pérdida de agua en la agricultura, especialmente en terrenos extensos, es un problema relevante que afecta la economía y el medio ambiente. El desgaste y la pérdida de agua en la agricultura son innecesarias y pueden atribuirse a factores como la pendiente del terreno y el abastecimiento de agua. La pendiente del terreno, por ejemplo, puede ocasionar una pérdida del 60% en la distribución del agua, lo que resulta en un descontrol en la cantidad de agua administrada en un determinado tiempo de riego. Además, la falta de control sobre el tiempo de riego puede resultar en aplicaciones insuficientes o excesivas de agua, lo que conlleva a pérdidas de tiempo, recursos hídricos y cultivos. La contribución del agua a la superficie terrestre se realiza a través de lluvias o riego, pero es importante tener en cuenta que la recolección del agua en la superficie no siempre beneficia a las plantas. Por lo tanto, es necesario intervenir para asegurar que las plantas reciban el agua de manera eficiente para su desarrollo.

Para abordar este problema, se propone el desarrollo e implementación de un software basado en la tecnología Arduino. Este software permitirá la automatización del riego

tecnificado por goteo, brindando un control preciso y programado del flujo de agua, asegurando que las plantas reciban la cantidad adecuada de agua en el momento oportuno. Con ello, se busca no solo mejorar la productividad agrícola, sino también contribuir a la conservación de recursos naturales al reducir el desperdicio de agua en el riego tecnificado. El sistema incorporará sensores de humedad y temperatura, lo que permitirá monitorear los niveles de humedad del suelo y prevenir el exceso o la falta de riego, mejorando así el crecimiento de las plantas. Por consiguiente, se considera dos aspectos fundamentales en la implementación: el diseño hidráulico y el diseño agronómico.

En cuanto al diseño hidráulico del sistema de riego tecnificado por goteo puede resultar tedioso debido a los cálculos necesarios para determinar las tuberías, accesorios, entre otros aspectos. Estos cálculos dependen de factores como la pendiente del terreno, la oferta y demanda de agua, entre otros. Por otro lado, el diseño agronómico para el sembrío de vegetales es esencial y puede calcularse manualmente o con software. Este cálculo está estrechamente relacionado con el diseño hidráulico, por lo que se realiza una comparación entre ambos para verificar si se compensan adecuadamente la oferta y la demanda de agua.

En síntesis, el problema que se aborda en este proyecto es la necesidad de mejorar la eficiencia del riego tecnificado por goteo mediante la automatización del proceso a través del desarrollo e implementación de un software basado en la tecnología Arduino. Esta solución permitirá un control preciso y programado del riego, optimizando el uso de agua y mejorando el crecimiento de las plantas.

## **1.2 Enunciado del Problema**

### **1.2.1 Problema General**

¿En qué medida la automatización de riego tecnificado por goteo mejora la eficiencia de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021?

### **1.2.2 Problemas específicos**



- ¿En qué medida la automatización de riego tecnificado por goteo mejora el tiempo de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021?
- ¿En qué medida la automatización de riego tecnificado por goteo mejora el consumo de agua mediante el desarrollo e implementación de un software con tecnología Arduino, 2021?
- ¿En qué medida la automatización de riego tecnificado por goteo mejora el nivel de humedad del suelo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021?

### 1.3 Justificación de la investigación

La presente investigación tiene como finalidad automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software basado en la tecnología Arduino y su justificación se fundamenta en la necesidad de disminuir el desperdicio de una gran cantidad de agua al momento que se riega por la poca eficiencia del riego. El riego tecnificado por goteo es una técnica ampliamente utilizada en la agricultura debido a su capacidad para proporcionar agua de manera precisa y controlada a las plantas. Sin embargo, el proceso de controlar manualmente el riego puede resultar laborioso y propenso a errores, lo que puede llevar a un uso ineficiente del agua y afectar negativamente el crecimiento de las plantas.

Según la Autoridad Nacional del Agua (ANA), en los procesos de diseño hidráulico y diseño agronómico, cerca de 13 millones de metros cúbicos de agua (MMC), que equivalen al 80% del volumen destinado a uso agrícola a nivel nacional, se pierden al año por deficiencias en el sistema de riego. Carlos Pagador Moya, jefe de la Autoridad Nacional del Agua, sostiene que esta situación es preocupante, ya que precisó que a nivel nacional se destinan 16.058 MMC por año al sector agrario y las pérdidas, que equivalen a 12.846 MMC, obedecen a deficiencias en el riego debido a factores de tiempo de riego, inadecuada distribución del diseño hidráulico. Por lo tanto, los sistemas que utilizan la tecnología Arduino ayudan a recoger información real del ambiente y procesar de forma real.

La implementación de esta solución tiene el potencial de optimizar el uso del agua y mejorar el crecimiento de los cultivos, lo que puede tener un impacto positivo tanto en la productividad agrícola como en la sostenibilidad ambiental.



## **CAPÍTULO II**

### **OBJETIVOS E HIPÓTESIS**

#### **2.1 Objetivos de la investigación**

##### **2.1.1 Objetivo general**

Automatizar el riego tecnificado por goteo para mejorar la eficiencia del riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

##### **2.1.2 Objetivo específicos**

- Determinar en qué medida la automatización de riego tecnificado por goteo mejora el tiempo de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021
- Determinar en qué medida la automatización de riego tecnificado por goteo mejora el consumo de agua mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.
- Determinar en qué medida la automatización de riego tecnificado por goteo mejora el nivel de humedad del suelo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

#### **2.2 Hipótesis de la Investigación**

##### **2.2.1 Hipótesis general**

La automatización de riego tecnificado por goteo mejora la eficiencia de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

##### **2.2.2 Hipótesis específicas**

- La automatización de riego tecnificado por goteo mejora el tiempo de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

- La automatización de riego tecnificado por goteo mejora el consumo de agua en el riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.
- La automatización de riego tecnificado por goteo mejora el nivel de humedad del suelo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

### 2.3 Operacionalización de variables

A. **Variable independiente:** Automatización de riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino.

B. **Variable dependiente:** Eficiencia de riego.

**Tabla 1 — Cuadro de operacionalización de variables**

VARIABLES	DIMENSIONES	INDICADORES	ÍNDICES/ESCALA
<b>Independiente</b> Automatización de riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino.	Eficiencia de desempeño del software	Comportamiento de tiempos	Porcentaje [ % ]
		Utilización de recursos	
<b>Dependiente</b> Eficiencia de riego	Tiempo	Tiempo de riego	Seg.
	Consumo de agua	Cantidad de agua	Litros/seg
	Humedad de suelo	Nivel de humedad	Porcentaje [ % ]



## CAPÍTULO III

### MARCO TEÓRICO REFERENCIAL

#### 3.1 Antecedentes

##### 3.1.1 Antecedentes internacionales

- a) ESCOBAR MANZABA, César Darío; FARFÁN ORELLANA, Karina Jazmín (2018), desarrolló la investigación titulada “*Diseño de un sistema de riego para la implementación de cultivos automatizados en el recinto playa seca del cantón el triunfo*”. Universidad Agraria del Ecuador. Cuyo objetivo fue: diseñar un sistema de riego por goteo, con la finalidad de optimizar el consumo del agua. El problema principal de la investigación fue el desperdicio del agua que se realizaba de manera manual. El estudio de la investigación tuvo un alcance del nivel descriptivo. Se creó un esquema detallado del área de riego del cultivo con el sistema de riego automatizado con la finalidad de conocer los componentes necesarios para su construcción, junto con un diseño general y vista aérea que posibilitó entender cómo sería el sistema de riego automatizado. Se concluye que, se logró el diseño del sistema de riego para la implementación de cultivos automatizados y que, para facilitar la construcción del sistema, también se elaboró el esquema de circuito de la placa Arduino con sus componentes principales y sensores.
  
- b) CASTILLO MELGAR, Carlos Alberto (2021) desarrollo la tesis “Diseño de un sistema de riego automatizado para cultivos de ciclo corto con Arduino. estudio de caso pimiento”. Universidad Estatal Península de Santa Elena, Ecuador. Cuyo objetivo fue: “Diseñar un sistema automatizado de riego por goteo para el cultivo de pimiento utilizando Arduino uno y tinkercad en función de la humedad del suelo” (p. 4). El problema de la investigación radica en el desaprovechamiento del recurso hídrico (agua), bajo rendimiento en el cultivo, falta de tiempo en la



labor de los agricultores, y pérdidas económicas en la producción final. Se analizó el uso de tecnologías aplicadas, como Arduino Uno, que permiten programar y tecnificar los tiempos de riego ideales y el caudal necesario para el buen desempeño del cultivo de pimiento. Concluyendo que, se diseñó satisfactoriamente el sistema automatizado de riego por goteo para el cultivo de pimiento, utilizando Arduino uno. Se empleó “12 sensores de humedad en las cuatro parcelas de la hectárea de cultivo de pimiento, en función de sus lecturas y del tipo de textura del suelo se activan o desactivas las electroválvulas para suministrar o no agua al mismo” (p. 30).

- c) En la investigación, CAÑÓN QUIROGA, Diana; CIFUENTES CAMACHO, Edna Luz (2018). Prototipo de un sistema automatizado de riego para jardines. Fundación Universitaria Los libertadores, Bogotá. Planteó el objetivo: “Diseñar e implementar un prototipo de sistema Automatizado de Riego para jardines” (p. 13). La presencia y uso de del sistema permitió controlar la cantidad y el tiempo utilizado para el riego de las plantas, facilitó las tareas que deben de realizarse en el riego de plantas. Aplicando una metodología descriptiva y de desarrollo del sistema la metodología ágil SCRUM. Se concluye que, se logró diseñar una “estructura base para la implementación del prototipo haciendo uso de Arduino, se implementó un prototipado con sensores y circuitos electrónicos. Se elaboró una aplicación de software que se integra con Arduino para la automatización del sistema de riego” (p. 57).
- d) SODESSA SOMA Shonkora *et al.* (2021), publicó el artículo “Modelado e Implementación de un Sistema Automático Sistema de riego por goteo”. University of Prince Edward Island, Canadá. Cuyo objetivo fue: montar un sistema de riego por goteo automático. El sistema propuesto utiliza sensores para monitorear la humedad del suelo y la temperatura utilizando la información del microcontrolador. Los sensores de humedad y temperatura del suelo Se usaron para detectar las condiciones climáticas que se enviaron al microcontrolador. Usamos un microcontrolador barato (PIC16F877A) conectado a varios sensores y actuadores para implementar el sistema. El software Proteus se utilizó para crear una interfaz gráfica de usuario (GUI) para monitorear el funcionamiento



del sistema de riego. El sistema envía automáticamente un mensaje al sistema operador, que monitorea el funcionamiento del sistema desde una pantalla de cristal líquido (LCD), cuando hay un sistema defectuoso. Concluyendo que: el sistema es capaz de proporcionar el suministro de agua necesario en épocas de condiciones climáticas adversas.

### 3.1.2 Antecedentes nacionales

- a) VILCHEZ CHUMPITAS Hans Aaron (2020). En la tesis titulada “Diseño e implementación de un sistema de riego por goteo automatizado para una hectárea de cultivo de durazno en el distrito de la Isla - Asia, Cañete”. El objetivo de esta investigación fue determinar cómo el diseño e implementación de un sistema de riego por goteo automatizado influyeron en la optimización del recurso hídrico en una hectárea de cultivo de duraznos en el distrito de Isla-Asia, Cañete. Los agricultores de esta región árida enfrentan una falta de agua debido al uso inadecuado de este recurso. La mayoría de ellos usan riego por surcos, lo que resulta en una falta de agua efectiva. En esta investigación, sugerimos complementar el riego por goteo, que es un tipo de riego altamente eficiente, con la automatización del riego. Para lograrlo, utilizamos un controlador Arduino Mega 2560 y una interfaz de Visual Studio. Esto nos permitió lograr un tipo de riego aún más eficiente. Para preservar este líquido esencial en la zona de estudio, los agricultores de la región pueden obtener una calidad superior de durazno y ahorrar dinero al reducir el uso de agua en menos del 6% en comparación con el riego por surco.
  
- b) CHANG WONG Leonel Irwing (2020). En el estudio “Diseño de un sistema automatizado de riego por goteo para aumentar la producción de Maíz en la Hacienda Durand”. Universidad Católica Santo Toribio de Mogrovejo, Chiclayo. El objetivo fue: “Diseñar un sistema automatizado de riego por goteo para aumentar la producción de maíz en la Hacienda Durand” (p. 18). Se utilizó el riego por inundación con dos motobombas de 4 HP, en 8 hectáreas de cultivo, de las cuales solo se trabajaron 3. Se calcularon las necesidades de orden hídrico por hectárea de maíz utilizando el software libre Cropwat 8.0 de la "Organización de las Naciones Unidas para la Alimentación y la Agricultura", lo que resultó en un caudal de 6.481 l/s, y se calculó la altura manométrica utilizando la ecuación

de Hazen-Williams. Estos dos parámetros fueron necesarios para determinar que se requerían dos electrobombas trifásicas con especificaciones técnicas similares a las de la electrobomba Speroni modelo 2C 40/180B de 10 HP, una electrobomba trifásica con especificaciones técnicas similares a la electrobomba Pedrollo modelo HF30A de 10 HP y una electrobomba trifásica con especificaciones técnicas similares a la electrobomba Pedrollo modelo CP 100 de 0.33 HP. Se utilizó un PLC de especificaciones similares al modelo SR3B261FU de Schneider Electric para automatizar el sistema de riego por goteo. El PLC se programó con el software Zelio Soft 2, lo que resultó en una interfaz hombre-máquina fácil de manejar. Esto permitió al cultivo recibir los recursos hídricos y nutrientes necesarios según su etapa de cosecha del maíz, lo que aumentó la producción de maíz y redujo las horas de trabajo en el campo y laboríos. Se llevó a cabo una evaluación económica del sistema automatizado de riego por goteo durante un período de diez años, en la que se analizaron los costos de la inversión inicial, el transporte, la instalación, el mantenimiento, el consumo de energía comercial, el pago a los trabajadores, el pago a la junta de usuarios, los costos de semillas y fertilizantes, así como la producción de maíz en las 8 hectáreas (p. 100).

- c) Tesis, HUAMÁN LIZANA, Kenji Franco; HUAMÁN COLUNCHE, José Wilder (2018). “*Diseño de un sistema automático de riego tecnificado para las áreas verdes de la Escuela de Ingeniería Electrónica y Facultad de Ciencias Físicas y Matemáticas–UNPRG*”. El objetivo principal de esta investigación fue: diseñar un sistema automático para regar de manera técnica todas las áreas verdes de la escuela de ingeniería electrónica y facultad de ciencias físicas y matemáticas de una universidad. Ya propuesto este trabajo porque considera aspectos de medición de áreas de perímetros de aquellas zonas que necesitan un riego para poder determinar los lugares donde funcionarían los aspersores, tuvo en cuenta la longitud y diámetro de los tubos, forma se consideró la cantidad de agua que debe contener el pozo. Posteriormente Se realizaron cálculos sobre el caudal de los aspersores además de las pérdidas que tenían las tuberías de presión cómase té en el variador de velocidad para evitar que exista interrupción en el control de riego. Para este efecto se ha realizado un programa consecutivo de



diagrama de escalera.(LADDER) en un PLC Schneider M340, Para el estudio se tomaron en cuenta 22 variables de entradas discreta y 39 variables de salida discreta, los procesos que se utilizaron para comunicarse con el variador de velocidad altivar 31 se nombraron MODBUS, El funcionamiento de este sistema se hizo desde un tablero de control que se ubicó en una caseta al costado de la FACFyM. En el proceso de funcionamiento el programa establecido se iba poniendo en funcionamiento de manera consecutiva con las válvulas que se adecuaron en base a las áreas que se tenían que regar, ADN se ha tomado en cuenta la longitud de la tubería para poder disponer los aspersores correspondientes. Se concluye que, el diseñar el sistema de riego tecnificado ha permitido mejorar las competencias que se han obtenido en los estudiantes dentro del área de automatización y control de la Escuela Profesional de Ingeniería Electrónica de la Universidad Nacional Pedro Ruiz Gallo.

- d) Tesis, QUISPE TAPARA, Hilcías. *Diseño de un sistema de riego automatizado por aspersión para viveros de café utilizando la tecnología Arduino en la empresa Viveros Ortiz-Pasco*; 2018. Universidad Católica los ángeles de Chimbote. El objetivo del estudio fue: Diseñar un sistema de riego automatizado por aspersión utilizando tecnología Arduino en la Empresa Vivero Ortiz-Pasco, que permitirá mejorar principalmente el tiempo de labor y el ahorro de agua. Metodológicamente la investigación está considerada como cuantitativa, el diseño fue no experimental del tipo descriptivo. Para el estudio se consideró una población de todos los trabajadores de la empresa, la muestra fue delimitada a 5 trabajadores, para recolectar información se tuvo en cuenta el cuestionario, en los resultados obtenidos a nivel de la dimensión de satisfacción se percibió que el 100% si acepta, además conforme con el uso de este sistema de riego, por otro lado sobre la dimensión necesidad de tomar en cuenta este sistema de riego se percibe que el 100% Si requiere tener conocimiento además de Modificar el sistema de riego. Se concluye que, “existe una necesidad de diseñar un sistema de riego automatizado que mejore el tiempo de trabajo y el ahorro de agua. Esta interpretación coincide con lo propuesto en la hipótesis general planteada en esta investigación en el que diseñar sistema de riego automatizado por aspersión utilizando la tecnología Arduino en la empresa Viveros Ortiz – Pasco; 2018,



mejorará el tiempo de trabajo y ahorro de agua” (p. 81).

### 3.1.3 Antecedente local

- a) Tesis, GUTIÉRREZ QUISPE, Roger David; RAMOS ENCISO, Leónidas. “Diseño del reservorio de concreto armado para el mejoramiento de riego en Comunidad de Vito Provincia Antabamba, Región Apurímac 2021”. Universidad Cesar Vallejo, Lima. El presente estudio tuvo como objetivo principal el diseño de una estructura hidráulica en este caso un reservorio de concreto armado ubicado en una ladera que es la cabecera de las parcelas de cultivo para mejorar o hacer más eficiente un sistema de riego tecnificado que puede ser por aspersión, micro aspersión y/o goteo. Para ello se iniciará los trabajos realizando un aforo de la fuente hídrica que nos otorgará el caudal de diseño (estudio hidrológico), levantamiento topográfico (estudio topográfico), diseño hidráulico donde se determinó las estructuras complementarias tales como desarenador, cerco perimétrico, canaleta de limpia, caja de válvula de alimentación al sistema de riego y caja de válvulas de limpia. Así como también el estudio geotécnico para con dichos resultados realizar el cálculo estructural. Se han utilizado equipos y/o instrumentos de observación directos y softwares tales como AutoCAD, Civil 3D, Excel y S10 para el procesamiento de la información tomada en campo y elaboración del presupuesto que costara la ejecución de dicha infraestructura. Instalar un sistema de riego tecnificado para una comunidad de esta zona alto andina generalmente es una inversión o proyecto financiado por el estado motivo por el cual el éxito de dicho proyecto depende de un buen diseño de la cámara de carga o reservorio ya que durante la fase del proceso constructivo sea por administración directa, por contrata a precios unitarios o a suma alzada lo diseñado por el Ingeniero proyectista debe ser lo más exacto posible en el replanteo de las medidas del largo ancho y profundidad del reservorio, así como también en los metrados y costos unitarios de las partidas deben coincidir con la realidad del campo. Se concluye que, “para reservorios ubicados en laderas se debe considerar la pantalla que va al cerro como muro de contención por lo que la sección debe ser trapezoidal. El reservorio cumple el papel más importante en todo un sistema de riego ya que nos garantiza la cantidad, la calidad del recurso hídrico y su operación” (p. 60).



## 3.2 Marco teórico

### 3.2.1 Riego por goteo

También conocido como riego localizado, en donde consiste en abastecer agua a los cultivos de manera continua, uniforme y lenta, aprovechando así evitamos encharcar el terreno y conservamos cierto grado de humedad necesaria en la tierra. El riego por goteo se coloca una vez que tenemos los bancales o terreno dispuesto para el cultivo y después sembramos o trasplantamos nuestras plantas. Se recomienda estén semienterradas las mangueras para oprimir la pérdida de agua por calor y viento o cubrirlas con el acolchado. Este tipo de riego puede ir desde el más franco y casero hecho con una manguera con agujeros finos que se la designa también cinta absorbente y poco más hasta los más sofisticados con programadores (que también los podemos hacer domésticos, dependiendo de la pericia de cada persona). Una particularidad de riego por goteo es el riego por exudación. Se diferencia del riego por goteo en que este tipo de riego se autorregula y sólo sale agua cuando el terreno va desperdiando humedad y se va secando, mientras que si la tierra ya está rociada no sale agua (SALINAS ARCOS, 2019).

- **Ventajas del riego por goteo**

El agua ya que circula superficialmente se ahorra en su cantidad usada. Además de no requerir el uso de energía y fuerza que se emplearía en un sistema adicional que bombee el agua. Bajos requerimientos de uso de energía y de potencia en el sistema de bombeo. Automatización completa de nuestro sistema de riego, con el programador podemos inspeccionar en todo momento el riego que queremos proporcionar en nuestro huerto. Disposición exacta del agua en el terreno en el que necesita la planta. Verdaderamente ponemos el agua sobre la raíz de la planta para que ella la distribuya. Fertirriego, es decir; uno de los progresos que dio el riego por goteo fue el registro de los nutrientes. Podemos administrar en el agua de riego los nutrientes precisos para nuestro cultivo, de esta manera mejoraremos el beneficio de nuestro huerto. Deducción de las malas hierbas. Al aplicar el agua directamente en la planta, lograremos evitar dar a las malas hierbas que crecen alrededor de nuestro cauce ya que el agua va verdaderamente donde queremos. Reducción de los inconvenientes de

plagas en las hojas. Debido a que el gotero emite el agua directamente sobre la raíz de la planta, no empapa a la hoja y evitamos el posible aumento de hongos. Por lo que asumimos que tener mucho cuidado con la colocación del gotero ya que si está muy unido al tronco puede generar enfermedades vasculares y más en plantas semileñosas. Lo podemos colocar en cualquier tipo de terreno. Como la mayoría de los elementos que se manejan son de plástico, se pueden moldear como queramos sobre cualquier espacio. Menor erosión del suelo. Como no circula agua sobre el cauce, la tierra no sufre erosión alguna como puede suceder. (SALINAS ARCOS, 2019)

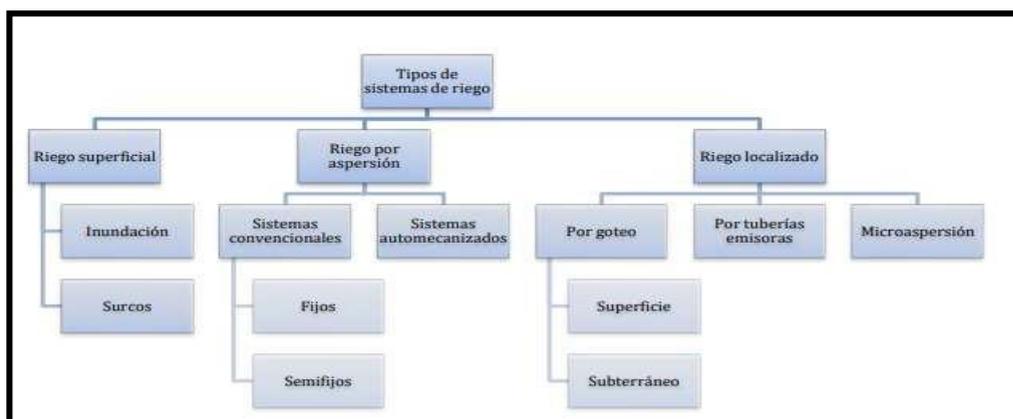
- **Desventajas del riego por goteo**

En comparación con otros sistemas de riego, puede ser más costoso en la instalación. Pero, si se tiene la posibilidad de colocar riego por goteo en el huerto es una buena opción. Ya que puede que el coste naciente sea más caro, pero a la larga económicamente y ecológicamente será más eficaz. (SALINAS ARCOS, 2019).

### 3.2.2 Sistema de riego

Según ESCOBAR MANZABA & FARFÁN ORELLANA (2018) afirma que “Un sistema de riego se denomina al conjunto de elementos físicos que hacen posible que un área determinada pueda ser cultivada con la aplicación de agua necesaria para las plantas” (p. 40).

Clasificación de los softwares de riegos.



FUENTE: Escobar Manzaba & Farfán Orellana , 2018

**Figura 1— Tipos de software de riegos**

### 3.2.3 Eficiencia de riego

Según CRUZ (2001) La eficiencia de riego se refiere a la capacidad de utilizar el agua de manera óptima para satisfacer las necesidades de los cultivos y minimizar el desperdicio, asegurarse de que sea consumida por los cultivos de manera adecuada.

Es importante tener en cuenta el tipo de cultivo y el sistema de riego utilizando para lograr un grado adecuado de eficiencia en el uso del agua.

Según ANTUNEZ, FELMER Y MORA(2009) sostiene que “La eficiencia con que los agricultores aplican el agua de riego depende de dos factores: el manejo del agua durante el riego, y las características hídricas del suelo que se está regando” (p. 73).

Estudios realizados sobre prototipos de riego automatizado o sistemas de riego inteligente con sensores y monitoreo automatizado de cultivo (CABRERA y PÉREZ, 2022; RENE, et al. 2022) utilizaron como dimensiones e indicadores para incrementar la eficiencia del riego: el tiempo de riego, frecuencia o consumo del riego y la humedad del suelo. Estos indicadores se medirán de acuerdo a los requerimientos de agua del cultivo.

- **Tiempo de Riego**

Se refiere a la duración y frecuencia del riego. Establecer programas de riego que se alineen con las necesidades específicas de las plantas y las condiciones climáticas que contribuye a la eficiencia. Según ANTUNEZ, et al.(2009) afirma que “entre los factores de manejo que influyen en la eficiencia de riego se encuentran el diseño del sistema, los caudales de riego utilizados, la frecuencia y el tiempo de riego empleado” (p. 74).

Según YARA (2024) El tiempo de riego es crucial para maximizar la eficiencia del riego por goteo, ya que permite ajustar la cantidad de agua que se aplica a las plantas. Un riego bien programado asegura que las plantas reciban la cantidad adecuada de agua, evitando tanto el exceso como la escasez. Es por ello que, El tiempo de riego debe ser ajustado para asegurar que las plantas reciban la cantidad adecuada de agua. Un riego demasiado corto puede



resultar en una hidratación insuficiente, mientras que un riego excesivo puede causar escorrentía y pérdida de nutrientes. La duración óptima depende de las necesidades hídricas del cultivo y de las condiciones del suelo.

Diferentes cultivos tienen diferentes requerimientos hídricos, y estos pueden variar a lo largo de las distintas etapas de crecimiento. Por lo tanto, el tiempo de riego debe ser adaptado a las necesidades específicas de cada cultivo en cada fase de su desarrollo (MENDOZA, 2013).

- **Consumo de agua**

Se refiere a la aplicación de la cantidad justa de agua necesaria para satisfacer las demandas hídricas de las plantas. Evitar el riego excesivo o insuficiente contribuye a la eficiencia general del sistema. Mide la cantidad de agua utilizada por las plantas (ANTUNEZ, et al., 2009).

La Eficiencia del riego se ve afectada por las pérdidas durante el riego. Estas pueden ser mínimas en sistemas bien diseñados y mantenidos, donde el agua se aplica directamente a la zona radicular, minimizando la evaporación y la escorrentía. En cambio, un mal diseño o mantenimiento puede resultar en una disminución significativa de la eficiencia (MENDOZA, 2013). Un factor crítico en el consumo de agua es la uniformidad en la distribución del agua. Un sistema de riego por goteo bien diseñado asegura que todas las plantas reciban una cantidad similar de agua, lo que es fundamental para maximizar la eficiencia. La evaluación de la uniformidad de descarga de los goteros es esencial para identificar y corregir problemas que puedan afectar la eficiencia del riego (MENDOZA, 2013). Por lo que, La posibilidad de medir y controlar la cantidad de agua aplicada permite ajustes en tiempo real, lo que contribuye a una mejor gestión del recurso hídrico. Esto incluye el uso de tecnologías avanzadas que pueden optimizar el riego y reducir el consumo de agua (MENDOZA, 2013).

- **Nivel de humedad del suelo**

Implica mantener el suelo en un rango óptimo de humedad para favorecer el crecimiento de las plantas. Monitorear y ajustar el riego según las condiciones del suelo garantiza que las raíces tengan acceso al agua necesaria. Mide que



porcentaje del agua aplicada se almacena en la zona radicular y está disponible para las plantas, se utiliza sensores de humedad del suelo en diferentes profundidades (ANTUNEZ, et al., 2009).

Diferentes estudios consideran a la “humedad del suelo” como indicador importante de la eficiencia del sistema de riego automatizado, ya que es el factor determinante para activar la irrigación del suelo. De acuerdo a la Organización de las Naciones Unidas para la Alimentación (FAO), esta se define como: “porcentaje de agua en el suelo (en peso). Normalmente se considera la cantidad de agua presente en el suelo entre el punto de marchitamiento y la capacidad de campo” (RENE, et al. 2022, p.12).

La capacidad del suelo para retener agua es fundamental para la eficiencia del riego por goteo. Suelos con alta capacidad de retención (como los arcillosos) pueden mantener la humedad por más tiempo, lo que permite intervalos de riego más largos. En contraste, suelos arenosos requieren riegos más frecuentes y cortos para evitar que la humedad descienda por debajo del nivel óptimo. El uso de dispositivos de medición de humedad permite a los agricultores evaluar la cantidad de agua disponible en el suelo. Esto ayuda a determinar el momento adecuado para regar y a ajustar la duración del riego para mantener la humedad dentro de un rango óptimo (MENDOZA, 2013).

### **3.2.4 Arduino**

#### **3.2.4.1 Microcontrolador Arduino-ATmega 328**

El Arduino Uno es una placa de microcontrolador de código abierto basada en el ATmega328. Tiene 14 pines de entrada/salida digital (de los cuales 6 se pueden utilizar como salidas PWM) y 6 entradas analógicas. Contiene todo lo necesario para soportar el microcontrolador, y puede ser simplemente conéctelo a una computadora con un cable Universal Serial Bus (USB) para comenzar.

El Arduino Uno se puede programar con el entorno de desarrollo integrado (IDE) de Arduino. El simple basado en C, el código del programa para Arduino se conoce como boceto. La colección de bocetos para funcionalidades específicas se remite a las bibliotecas.

El Arduino se puede programar hasta 32 KB de memoria. Arduino puede funcionar de forma autónoma sin estar conectado a una computadora, o alternativamente programado para responder principalmente a los comandos enviados desde la computadora a través de varias interfaces de software o a los datos adquiridos de los canales de entrada. El Arduino UNO basado en el Microcontrolador ATmega-328. (LATHA, MURTHY, & KUMAR, 2016)

Según TORRENTE (2013), menciona que Arduino tiene tres componentes fundamentales:

- **Una placa hardware libre** que contiene un microcontrolador reprogramable y una serie de pines-hembra (los cuales están unidos internamente a las patillas de E/S del microcontrolador) que nos dan la facilidad de incorporar allí de manera fácil y rápida distintos sensores y actuadores. Por otro lado, cuando hablamos sobre “placa Arduino”, se tiene que especificar el modelo indicado. Por qué existen diversos tipos de placas Arduino y cada uno de ellos tienen una característica distinta tales como: “tamaño físico, número de pines-hembra ofrecidos, modelo de microcontrolador incorporado” y respuesta se tiene, la cantidad de memoria que se puede utilizar. Pero su funcionamiento en realidad es bastante parecido entre sí.
- **SOFTWARE** (o también “un sistema integrado”) gratis, libre y multiplataforma (ya que funciona en diferentes sistemas operativos como Linux, MacOS y Windows) que debemos instalar en una laptop o pc y que nos admite escribir, verificar y guardar (“cargar”) en la memoria del microcontrolador de la placa Arduino el conjunto de instrucciones que queremos que este comience a ejecutar. Dicho de otra manera, nos permite programarlo. Una vez desarrollado las líneas de código en su microcontrolador, la placa no necesariamente debe de estar conectada a ninguna laptop o PC ya que este puede funcionar autónomamente si se conecta a una fuente de alimentación.



- **Lenguaje de programación libre.** Se entiende por “lenguaje de programación” cualquier idioma artificial el cual este diseñado para interpretar líneas de código o instrucciones (siguiendo unas determinadas reglas sintácticas) que pueden ser procesadas por máquinas. exactamente dentro del lenguaje Arduino, hallamos elementos semejantes a muchos otros lenguajes de programación que hay en el mercado, así como también distintos comandos también llamados “instrucciones” y/o “ordenes” que nos permiten dar órdenes exactas y sin errores los comandos son instrucciones exactas que queremos ordenar en el microcontrolador de la placa. Estos comandos los escribimos en código C++ en el Arduino. Viene a ser tanto en un contexto que permite desarrollar un lenguaje de programación Arduino donde se imagina otros entornos de lenguaje libre que ya existen previamente.



FUENTE: Caicedo, 2017

**Figura 2 — Placa de Arduino Uno**

#### 3.2.4.2 Memorias del Arduino

Menciona TORRENTE (2013) , que otra situación que se tiene que conocer de los microcontroladores son los tipos y la cantidad de memoria que estos puedan almacenar en su interior. En el caso del ATmega328P tenemos:

- **Memoria Flash:** Es la memoria persistente en el cual se almacena eventualmente, las líneas de código que se “ejecuta el

microcontrolador (hasta una nueva reescritura si se da el caso). En el caso del ATmega328P tiene una capacidad de 32KB”.

- **Memoria SRAM:** memoria volátil donde se guarda información en un momento que se precisa en el programa (grabado separadamente en la memoria Flash, recordemos) necesita crear o manipular para el correcto funcionamiento. Estos datos suelen tener un contenido variado a lo largo del tiempo de ejecución del programa y cada uno es de un tipo concreto (es decir, un dato puede contener un valor numérico entero, otro un número decimal, otro un valor de tipo carácter... también pueden ser cadenas de texto fijas u otros tipos de datos más especiales). Independientemente del tipo de dato, su valor siempre será eliminado cuando se deje de dar eléctricamente al microcontrolador. En el caso del ATmega328P esta memoria tiene una capacidad de 2KB.
- **Memoria EEPROM:** memoria persistente donde se almacenan datos que se desea que permanezcan grabados una vez apagado el microcontrolador para poderlos usar posteriormente en siguientes reinicios. En el caso del ATmega328P esta memoria tiene una capacidad de 1 KB, por lo que se puede entender como una tabla de 1024 posiciones de un byte cada una.

### 3.2.5 Protocolos de Comunicación

“El microcontrolador, a través de algunos de sus pines de E/S, utiliza el sistema de comunicación serie para transmitir y recibir órdenes y datos hacia/desde otros componentes electrónicos. Esto es debido sobre todo a que en una comunicación serie solo se necesita en teoría un único canal (un único “cable”), mientras que en una comunicación en paralelo se necesitan varios cables, con el correspondiente incremento de complejidad, tamaño y coste del circuito resultante”. En este sentido, los estándares más importantes son: (TORRENTE, 2013).

- **I2C (Inter-Integrated Circuit, también conocido con el nombre de TWI –de “Two-Wire”, literalmente “dos cables” en inglés–):** Es un sistema que se usa

bastante dentro de las comunicaciones a través de los circuitos integrados entre sí. Su característica fundamental indica que se realizan dos líneas para emitir la información: una (llamada línea “SDA”) apoya a conducir los datos (los 0s y los 1s) y otra (llamada línea “SCL”) se utiliza para remitir señal de reloj. En realidad, se requerirían dos líneas más: la de tierra común y la de alimentación, pero estas ya se consideran como existentes en el circuito.

- **SPI (Serial Peripheral Interface):** Así como el sistema I<sup>2</sup>C, el sistema de comunicación SPI está considerado como “un estándar que permite controlar (a cortas distancias) prácticamente cualquier dispositivo electrónico digital que acepte un flujo de bits serie sincronizado (es decir, regulado por un reloj). Igualmente, un dispositivo conectado al bus SPI puede ser “maestro” en inglés, “master” o “esclavo” en inglés, “slave”, en el que el primero es quien inicia la transmisión de datos y también genera la señal de reloj (aunque, como con I<sup>2</sup>C, con SPI tampoco es necesario que el maestro sea siempre el mismo dispositivo) y el segundo se limita a responder”.

### 3.2.6 Alimentación

La cantidad de energía que necesitará la placa Arduino (incluyendo el microcontrolador y el resto de componentes) es de 5 V. Para lograr esta carga de energía se puede realizar de diferentes formas (TORRENTE ARTERO, 2013):

- **Conectando la placa Arduino a una fuente externa,** tal como un adaptador AC/DC o una pila. “Para el primer caso, la placa cuenta con un zócalo donde se conecta una clavija de 2,1 milímetros de tipo “jack”. Para el segundo, los cables sobresalientes de los bornes de la pila se conectan a los pines-hembra marcados como Vin y Gnd (positivo y negativo respectivamente) en la zona de la placa marcada con la etiqueta POWER”.
- **Conectando la placa Arduino a nuestro computador** A través de un cable USB. Para esto la placa tiene un conector USB hembra de tipo B. La alimentación eléctrica + obtenida de esta forma se ajusta de manera frecuente a los 5 V de trabajo y ofrece un máximo de 500 mA de corriente

(por lo tanto, la potencia consumida por la placa es en ese caso de unos 2,5 W). Si en algún momento por el conector USB pasa mayor intensidad de lo necesario, la placa Arduino se protege por un polifusible reseteable que de manera automática quiebra la conexión hasta que la situación eléctrica vuelva a lo normal.

### 3.2.7 Componentes del Arduino

(CAICEDO, 2017) Existen múltiples variantes del Arduino. En este caso, usaremos el Arduino UNO que es el más común.

- **Potencia - USB (1) / Conector de Adaptador (2).** Cada placa Arduino requiere de una forma de estar alimentado eléctricamente. Esta puede ser alimentado desde un cable USB que viene de su ordenador o un cable de corriente eléctrica con su respectivo adaptador.
- **Pines (5V, 3.3V, GND, Analog, Digital, PWM, AREF).** Los pines en la placa Arduino son los que se conectan a los cables de un circuito. El Arduino consta de variados tipos de ingresos y cada uno de ellos están señalados en el tablero y se usan para distintas funciones.
- **GND (3):** Abreviatura de "tierra" (en Inglés). Existen diversos pines GND en el Arduino, cualquiera de ellos se puede usar para realizar la conexión del circuito a la tierra.
- **5V (4) y 3.3V (5).** Son los suministros pin 5V 5 voltios de energía, y los suministros de pin 3.3V 3.3 voltios de potencia.
- **Analógico (6).** El área de pines en el marco del 'analógica' etiqueta (A0 a A5) son analógicas. Estos pines hacen lectura a las señales que emite el sensor analógico (como un sensor de temperatura) y convertirlo en un valor digital que se puede leer.
- **Digital (7).** Son los pines digitales (del 0 al 13). Estos pines se utilizan para la entrada digital (como decir, si se oprime un botón) y salida digital (como encender un LED).
- **PWM (8).** Existe una señal de (~) al lado de algunos de los pines digitales (3, 5, 6, 9, 10 y 11). Estos pines funcionan como pines digitales normales, sin embargo, se utilizan como moduladores del ancho de los pulsos (PWM,



por sus siglas en Ingles).

- **AREF (9).** Soportes de referencia analógica. En el mayor de los casos se dejan estos pines, solos. A veces se usa para establecer una tensión de referencia externa (entre 0 y 5 voltios) como el límite superior para los pines de entrada analógica.
- **Botón de reinicio (10).** Presionando el botón se conectará parcialmente el pin de reset a tierra y vuelve a iniciar cualquier código que se carga en el Arduino. Esto puede ser muy útil si el código no se repite, pero quiere probarlo varias veces.
- **Indicador LED de alimentación (11).** Este LED debe prenderse siempre que se haga una conexión a la placa Arduino a una toma eléctrica. Si esta luz no prende, hay una alta probabilidad de que algo falla.
- **LEDs RX TX (12).** TX es la abreviatura de transmisión, RX que significa recepcionar. Estas marcas brotan en la electrónica para indicar los pasadores que se responsabilizande informar en serie. En este caso, hay dos lugares en la Arduino UNO donde aparecen TX y RX - una vez por pines digitales 0 y 1, y por segunda vez junto a los indicadores LED de TX y RX (12). Estos LEDs brindan indicadores favorables visuales que permitirán al Arduino recibir y trasmitir datos.
- **Microcontrolador (13).** Lo negro con todas las patas de metal es un circuito integrado (IC, por sus siglas en Ingles). Se considera como el cerebro del Arduino. La principal IC en el Arduino es un poco distinto al tipo de placa a placa tipo, Sin embargo, esto se da por la línea de ATmega de CI de la empresa ATMEL. Esta acción es fundamental porque conduce a la necesidad de estar presente el tipo IC (junto con su tipo de tarjeta) antes de subir un nuevo programa desde el software de Arduino. Esta información se puede hallar en la escritura en la parte superior de la IC. Si se quiere conocer más al detalle sobre las diferencias entre los distintos circuitos integrados, la lectura de las hojas de datos tiene a ser una buena opción.
- **Regulador de Voltaje (14).** -Esto no es realmente algo que se puede (o debe) interactuar con el Arduino. “Pero es potencialmente útil para conocer que se encuentray para qué sirve. El regulador de voltaje realiza lo que se indica – lleva a cabo el controlde la cantidad de tensión que se deja en la





La norma ISO/IEC 25000 promueve una evaluación integral de la calidad del software, sin embargo, la evaluación puede ser más dirigida y efectiva, ajustada a las necesidades específicas del entorno en el que se implementa el software (Balseca, 2014). En ese sentido, solo se ha de considerar la eficiencia del desempeño del software en un contexto específico donde el rendimiento y la capacidad de gestión son cruciales para el éxito del sistema. La eficiencia del desempeño es un aspecto fundamental que impacta directamente en la experiencia del usuario y la operatividad del sistema asociados a la infraestructura tecnológica.

Para medir la eficiencia del desempeño, se consideran las siguientes características que puede depender del contexto y los requisitos particulares del sistema que se esté evaluando.

- **Comportamiento de tiempos:** Se refiere al tiempo que tarda un sistema en responder a una solicitud o realizar una operación. Se puede medir utilizando herramientas de monitoreo o pruebas de rendimiento. Un buen rendimiento se refleja en tiempos de respuesta rápidos y consistentes (MARIN, 2021).

Las métricas a medir son:

- **Tiempo de respuesta** se refiere al tiempo estimado necesario para completar una tarea determinada (ISO 25000, 2020).
- **Tiempo de espera** se refiere en completar un trabajo con el sistema.
- **Rendimiento** se refiere a la cantidad de tareas que pueden ser procesadas (ISO 25000, 2020).

- **Utilización de recursos:** Se refiere a la cantidad de recursos del sistema, como CPU, memoria o ancho de banda de red, utilizados durante la ejecución. Se puede medir mediante el monitoreo del uso de recursos del sistema. Un software eficiente debería utilizar estos recursos de manera óptima para evitar sobrecargas y garantizar un rendimiento estable (MARIN, 2021).

Las métricas a medir son:

- **Utilización de CPU** se refiere al tiempo que el CPU se utiliza para realizar una tarea (ISO 25010, 2018)



- **Utilización de la memoria** se refiere al espacio de memoria que se utiliza para realizar una tarea (ISO 25010, 2018)

### **Determinación de atributos de calidad externa según nivel de importancia**

Según BALSECA (2014), “se establece el grado o nivel de importancia para determinar los atributos de calidad que serán medidos” (p, 55)

- **Nivel Alto (A):** Este nivel de importancia indica que se debe realizar las mediciones de los atributos. El intervalo referencial de porcentaje de importancia va desde el 70% al 100%.
- **Nivel Medio (M):** En este nivel se puede o no realizar las mediciones de los atributos. El intervalo referencial de porcentaje de importancia va desde el 25% al 69%.
- **Nivel Bajo (B):** En este nivel no es de importancia realizar las mediciones de los atributos. El intervalo referencial de porcentaje de importancia va desde 1% al 24%.
- **No aplica (NA):** En este nivel no se pueden medir los atributos. El porcentaje de importancia es de 0%.

Asimismo, según lo expuesto por BALSECA (2014), Los niveles de puntuación final para la calidad externa del producto de software se representan de la siguiente manera:

- Muy satisfactorio, cuando el nivel de puntuación cumple con los requisitos y la escala de medición está entre 8.75 a 10.00 puntos.
- Satisfactorio, cuando el nivel de puntuación es aceptable y la escala de medición está entre 5 a 8.74 puntos.
- Insatisfactorio, cuando el nivel de puntuación es mínimamente aceptable o inaceptable y la escala de medición está entre 2.75 a 0 puntos.

## **3.3 Marco conceptual**

### **3.3.1 Actuador**

“Componente que regula la humedad de suelo al nivel requerido” (SALCEDO, 2014, p 35)



### **3.3.2 Evotranspiración**

Es el proceso mediante el cual el agua se evapora desde la superficie del suelo y se transpira por las plantas. Es una medida de la cantidad de agua que se pierde a través de la evaporación y la transpiración en un área determinada durante un período de tiempo específico. La evapotranspiración es un factor importante a considerar en la gestión del riego y en la estimación de las necesidades hídricas de los cultivos (SANTOS, 2010).

### **3.3.3 Diseño agronómico**

“Evalúa todos los datos necesarios para que la instalación de riego sea capaz de suministrar con eficiencia óptima el agua a los cultivos en periodos de máximas necesidades, consiguiendo humedecer el volumen del suelo suficiente para un desarrollo eficiente de las raíces y un efectivo control de sales” (DE LOS ANGELES, 2000, p. 55)

### **3.3.4 Diseño hidráulico**

“Define los diámetros y longitudes de las diferentes tuberías que componen el sistema (regantes, distribuidoras y conducción) bajo un criterio de optimización” (MEDINA, 2000)

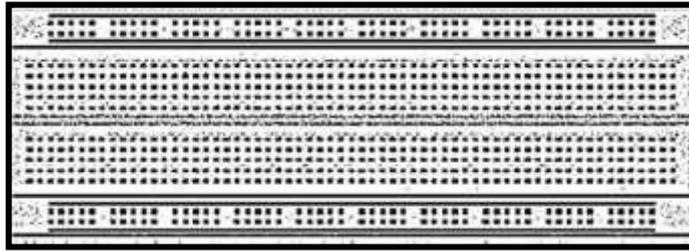
### **3.3.5 Relación agua-suelo-planta**

De acuerdo con (PALOMINO, 2019), las relaciones entre el agua, el suelo y las plantas incluyen la capacidad del suelo para retener agua y permanecer bien drenado, capacidad de campo, las características del flujo del agua en los suelos, las propiedades físicas del suelo como textura, estructura y propiedades químicas del suelo, incluyendo la concentración de sales solubles y nutrientes al movimiento, uso y evaporación del agua localizada en el suelo.

### **3.3.6 Placas de prototipado**

Conocidas como “protoboards”), las “perfboards” y las “stripboards”. Es una placa perforada con conexiones internas en la que podemos insertar las patas de componentes electrónicos tantas veces como queramos, realizando así las conexiones de circuitos sin tener la necesidad de llevar acabo una soldadura. La imagen que se presenta demuestra la apariencia hacia el exterior de una Breadboard típica, que no deja de ser un conjunto de filas con agujeros (TORRENTE, 2013).





FUENTE: Torrente, 2013

**Figura 4 — Placa de prototipado (protoboards)**

### 3.3.7 Sensor

Los sensores convierten diferentes estímulos físicos en señales eléctricas analógicas o digitales. Pueden clasificarse según el tipo de estímulo que miden, como acústicos, eléctricos, magnéticos, ópticos, térmicos y mecánicos. Estos sensores tienen la capacidad de medir una variedad de parámetros, incluyendo condiciones climáticas, características del suelo, calidad del agua y aspectos relacionados con los cultivos, como el crecimiento del tronco, la madurez de los frutos y el nivel de clorofila. (RAMÍREZ Y MAZON, 2018).

## CAPÍTULO IV METODOLOGÍA

### 4.1 Tipo y nivel de investigación:

#### 4.1.1 Tipo de investigación

Esta investigación corresponde a una investigación aplicada. BEHAR (2018), afirma que este tipo de investigación. Se caracteriza porque busca la aplicación o utilización de los conocimientos que se adquieren (p.20).

#### 4.1.2 Nivel de estudio:

El nivel de investigación que se utilizó en la presente investigación es explicativo. Según HERNÁNDEZ y otros (2014) afirma que dicha investigación está dirigida a responder por las causas de los eventos y fenómenos físicos o sociales; se enfoca en explicar por qué ocurre un fenómeno y en qué condiciones se manifiesta, o por qué se relacionan dos o más variables (p. 86).

### 4.2 Diseño de la investigación:

El diseño de investigación que se aplicó es Pre-experimental

Este tipo de diseño permite intervenir en el manejo de una variable denominado independiente, para obtener resultados en la otra variable dependiente. Según HERNÁNDEZ y otros (2014) “consiste en administrar un estímulo o tratamiento a un grupo y después aplicar una medición de una o más variables para observar cuál es el nivel del grupo en éstas; no hay manipulación de las variables o grupos de contraste” (p. 136).

Diagrama del diseño de investigación

## G 01 X 02

Donde:

**G** : Grupo experimental

**O<sub>1</sub>** : Reducción de la cantidad de agua

**X** : Implementación de sistema automatizado

**O<sub>2</sub>** : Reducción en del tiempo de gasto del agua

### 4.3 Población y muestra

#### 4.3.1 Población

La investigación se desarrolló en un área experimental de 77 m<sup>2</sup>, considerando que el área de riego tiene una anchura de 7m y una longitud de 11m. conformada por 11 surcos distanciados a 50 cm entre surcos, con 19 goteros.

#### 4.3.2 Muestra

La muestra del experimento está constituida de 11 surcos distanciados a 50 cm entre surcos, con 19 goteros. El área total del experimento es de 77 m<sup>2</sup>.

### 4.4 Procedimiento

a) **Etapa I:** Aprobación de proyecto de tesis.

b) **Etapa II:** Planificación, análisis y recopilación de la información.

- Se observó la problemática existente
- Se analizó la información obtenida y se obtiene los requerimientos que tendrán el sistema de riego tecnificado por goteo.

c) **Etapa III:** Diseño agronómico

En esta etapa se determinó la cantidad de agua para el cultivo de la remolacha o beterraga.

- Importación de las precipitaciones de la ciudad de Abancay con CLIMWAT.
- Importación de datos de precipitación, clima, etc.
- Duración de las etapas de crecimiento del cultivo para distintos períodos de siembra y regiones climáticas (días).
- Interfaz de la programación del cultivo de remolacha.

**d) Etapa IV: Diseño hidráulico**

En esta etapa para el riego tecnificado por goteo se determinó el diseño o estructura de las tuberías de riego en conjunto con los materiales correspondientes.

- Esquema del diseño hidráulico.
- Diseño de válvula en el Arduino.
- Diseño de DHT- sensor de temperatura en Arduino
- Esquema de sensor de humedad

**e) Etapa V: Desarrollo del software**

- Programación en Arduino UNO para el funcionamiento del riego por goteo

**f) Etapa VI: Interfaz del software con tecnología Arduino**

**g) Etapa VII: Integración de sensores e implementación de actuadores.**

**h) Etapa VIII: Implementación en el campo.**

**i) Etapa IX: Evaluación y análisis de resultados.**

**j) Etapa X: Redacción de informe final**

- Se realizó el informe final
- Se realizó las conclusiones y recomendaciones
- Presentación de informe final

## 4.5 Técnica e instrumentos

### 4.5.1 Técnica

La técnica utilizada es la observación.

### 4.5.2 Instrumentos

La experimentación se realizó al inicio y al final de la investigación utilizando los siguientes instrumentos:

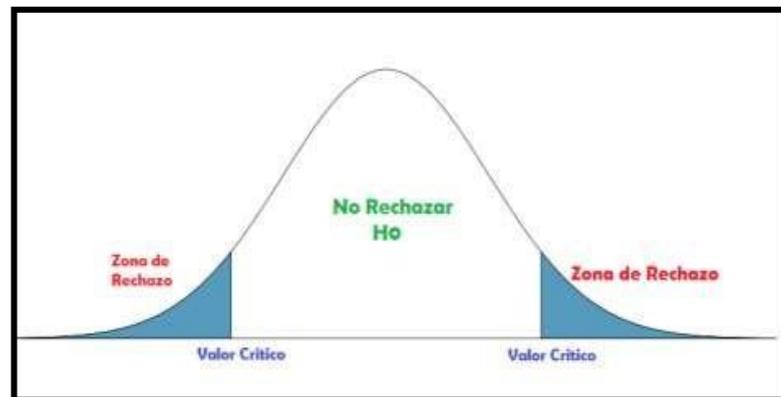
- Ficha de registro de datos
- Ficha de análisis

## 4.6 Análisis Estadístico

Para el procesamiento y análisis de datos se utilizó Excel 2019 y el paquete estadístico SPSS y Minitab, para luego ser interpretados de acuerdo al resultado obtenido.

Como la muestra  $n < 30$ , entonces se utilizó la distribución t-student. Con un nivel de

significancia  $\alpha = 5\%$ . En la Figura 5, se muestra la región crítica o regla de decisión para la prueba de hipótesis.



FUENTE: elaboración propia

**Figura 5 — Prueba de hipótesis**

## **CAPÍTULO V**

### **RESULTADOS Y DISCUSIONES**

#### **5.1 Análisis de resultados**

Los resultados obtenidos en el desarrollo de la presente investigación se detallan a continuación, a partir de los datos recopilados en la automatización del riego tecnificado por goteo mediante el desarrollo e implementación del software basado en la tecnología Arduino.

##### **5.1.1 Diseño agronómico para determinar la cantidad de agua para el cultivo**

El propósito del diseño agronómico es determinar el valor máximo de la evapotranspiración que puede cambiar con el tiempo. Por lo tanto, los sistemas de riego deben estar preparados para proporcionar suficiente agua a los cultivos cuando surjan necesidades críticas. En la presente investigación se está tomando como referencia el cultivo de remolachas o beterragas.

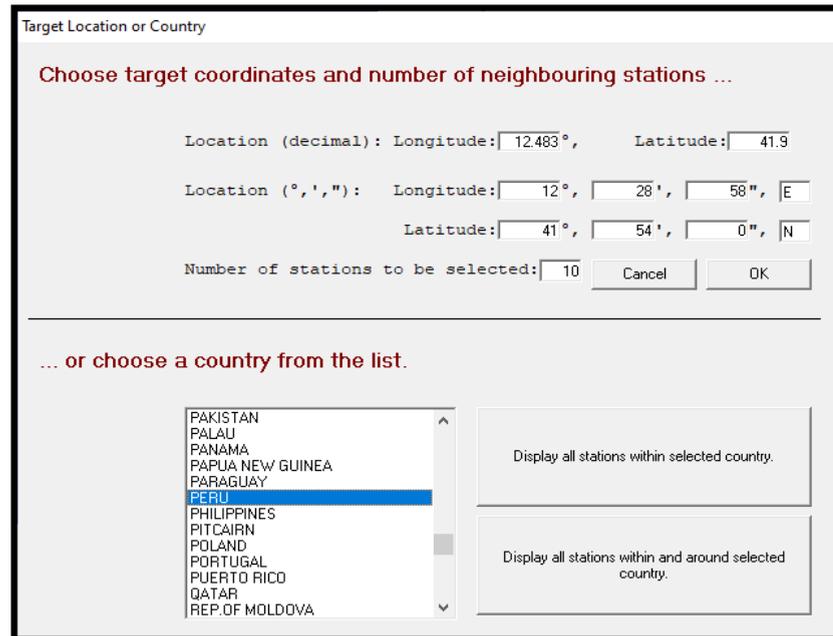
##### **A. Importancia de las precipitaciones de la ciudad de Abancay con CLIMWAT**

La fracción de la precipitación total utilizada es importante para satisfacer las necesidades de agua del cultivo.

Se usó el programa CLIMWAT 2.0, para la localización y exportación de precipitación meteorológicas de la ciudad de Abancay- Perú.

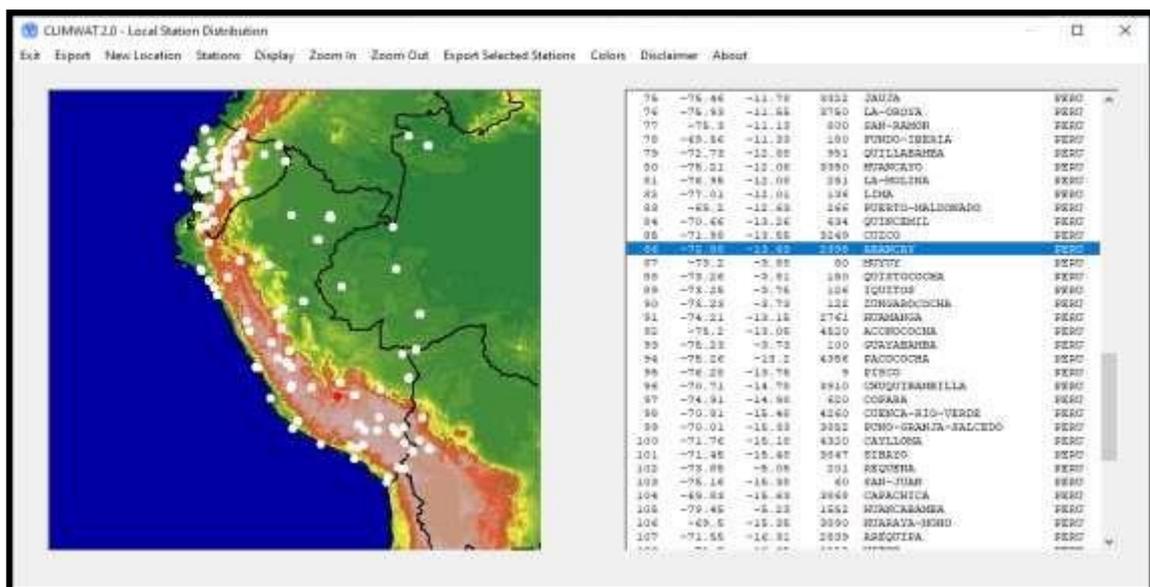
En la Figura 6, 7, 8 y 9 muestran cómo se exporta los datos de las precipitaciones y clima desde la ciudad de Abancay.

.



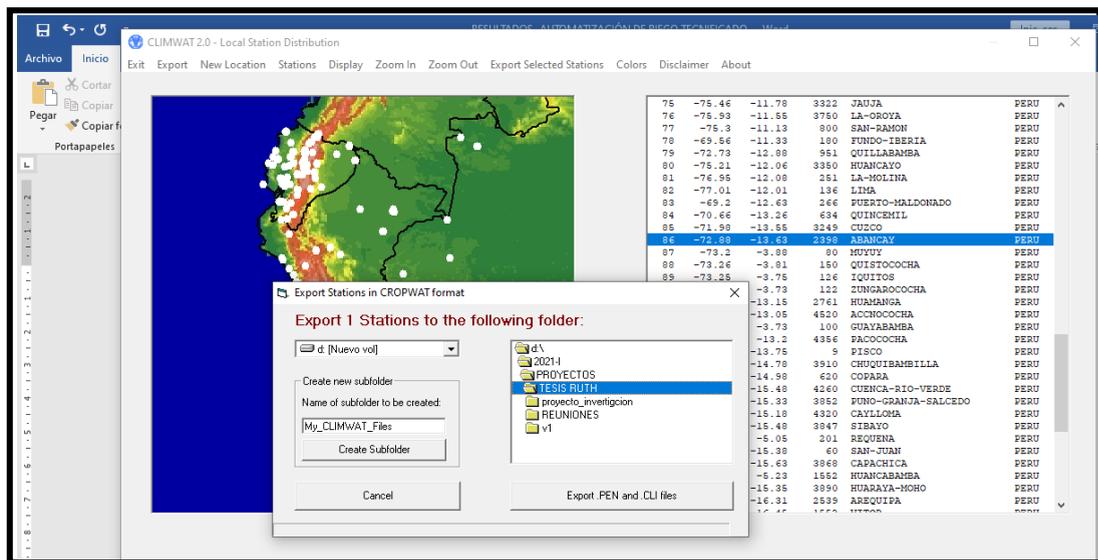
FUENTE: CLIMWAT 2.0

**Figura 6** — Ubicación del lugar donde se exportará los datos de precipitaciones y clima



FUENTE: CLIMWAT 2.0

**Figura 7** — Ubicación Abancay



FUENTE: CLIMWAT 2.0

Figura 8 — Exportación de datos de clima y precipitación

Day	Temp (°C)	Humidity (%)	Wind Speed (km/h)	Wind Dir (°)	Cloudiness (%)	Evapotranspiration (mm)	Precipitation (mm)
1	24.6	8.4	71.6	164.2	5.72	19.24	3.61
2	24.4	8.4	69.3	164.2	5.58	18.82	3.59
3	24.2	8.6	65.1	164.2	5.73	18.15	3.49
4	25.2	8.8	59.5	138.2	6.87	18.15	3.43
5	24.5	8.5	55.0	138.2	7.57	17.21	3.18
6	23.6	8.0	51.8	164.2	7.73	16.33	3.09
7	23.9	7.9	53.2	198.7	7.81	16.85	3.31
8	24.6	7.4	56.8	233.3	7.97	18.74	3.68
9	24.4	8.5	60.2	233.3	6.91	19.13	3.80
10	27.7	9.9	55.0	233.3	7.16	20.84	4.56
11	26.6	9.6	58.5	198.7	7.00	21.12	4.41
12	24.7	8.9	69.2	198.7	5.68	19.13	3.75
13							
14							

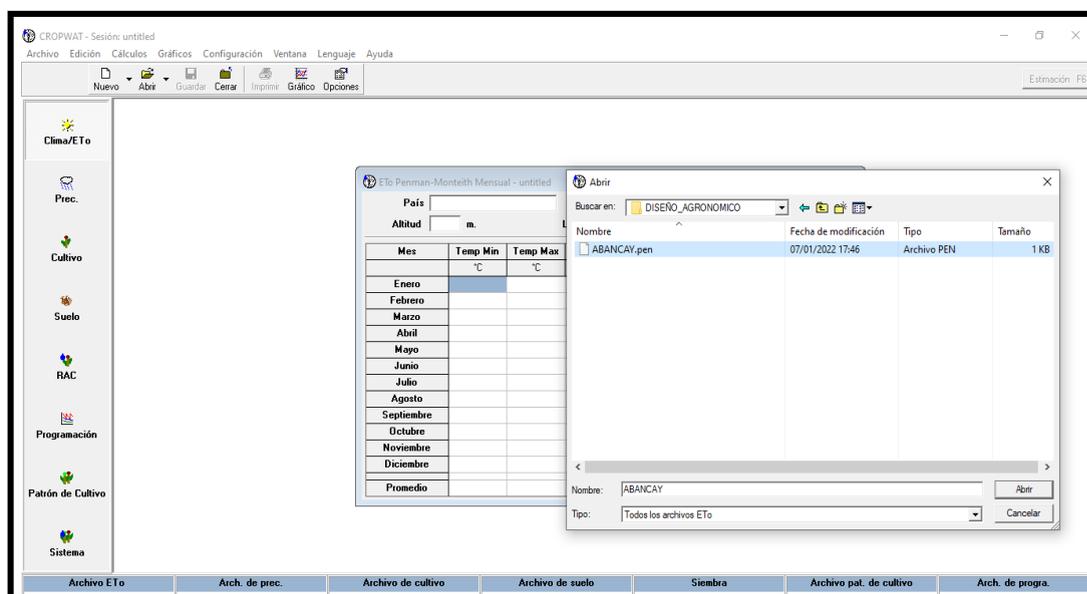
FUENTE: CLIMWAT 2.0

Figura 9 — Datos de la estación de ABANCAY

### B. Importación de datos de precipitación, clima, etc.

Para el cálculo de la evapotranspiración de cultivo de la remolacha o beterraga que se sembró. Se abre el archivo descargado por el software CLIMWAT, para importar estos datos se utilizará el software *CROPWAT*.

En las figuras 10, 11 y 12 se muestran los resultados de la precipitación efectiva obtenido por el software *CROPWAT*



CROPWAT - Sesión: untitled - [ETo Penman-Monteith Mensual - D:\2021-IPROYECTOS\TESIS RUTH\DISEÑO\_AGRONOMICO\ABANCAY.pen]

País PERU Estación ABANCAY

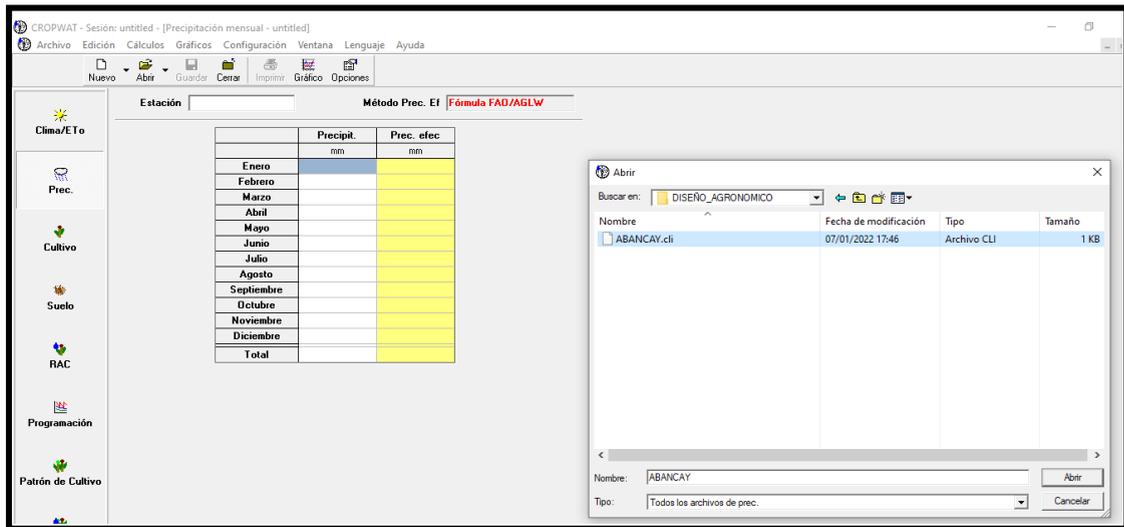
Altitud 2398 m. Latitud 13.63 °S Longitud 72.88 °W

Mes	Temp Min °C	Temp Max °C	Humedad %	Viento m/s	Insolación horas	Rad MJ/m²/día	ETo mm/día
Enero	8.4	24.6	72	1.9	5.7	19.2	4.05
Febrero	8.4	24.4	68	1.9	5.6	18.8	4.02
Marzo	8.6	24.2	65	1.9	5.7	18.2	3.91
Abril	8.8	25.2	60	1.6	6.9	18.2	3.84
Mayo	8.5	24.5	55	1.6	7.6	17.2	3.56
Junio	8.0	23.6	52	1.9	7.7	16.3	3.49
Julio	7.9	23.9	53	2.3	7.8	16.9	3.76
Agosto	7.4	24.6	57	2.7	8.0	18.8	4.26
Septiembre	8.5	24.4	60	2.7	6.9	19.2	4.33
Octubre	9.9	27.7	55	2.7	7.2	20.9	5.19
Noviembre	9.6	26.6	59	2.3	7.0	21.1	4.95
Diciembre	8.9	24.7	69	2.3	5.7	19.1	4.22
Promedio	8.6	24.9	60	2.2	6.8	18.7	4.13

FUENTE: CROPWAT 8.0

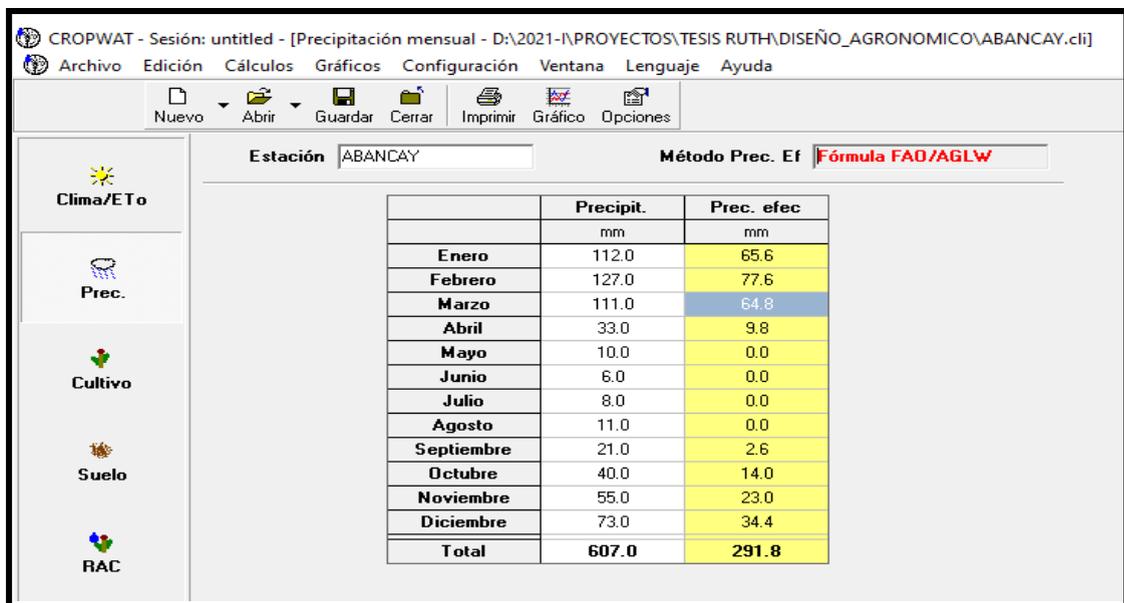
Figura 10 —Método FAO Penman Monteith





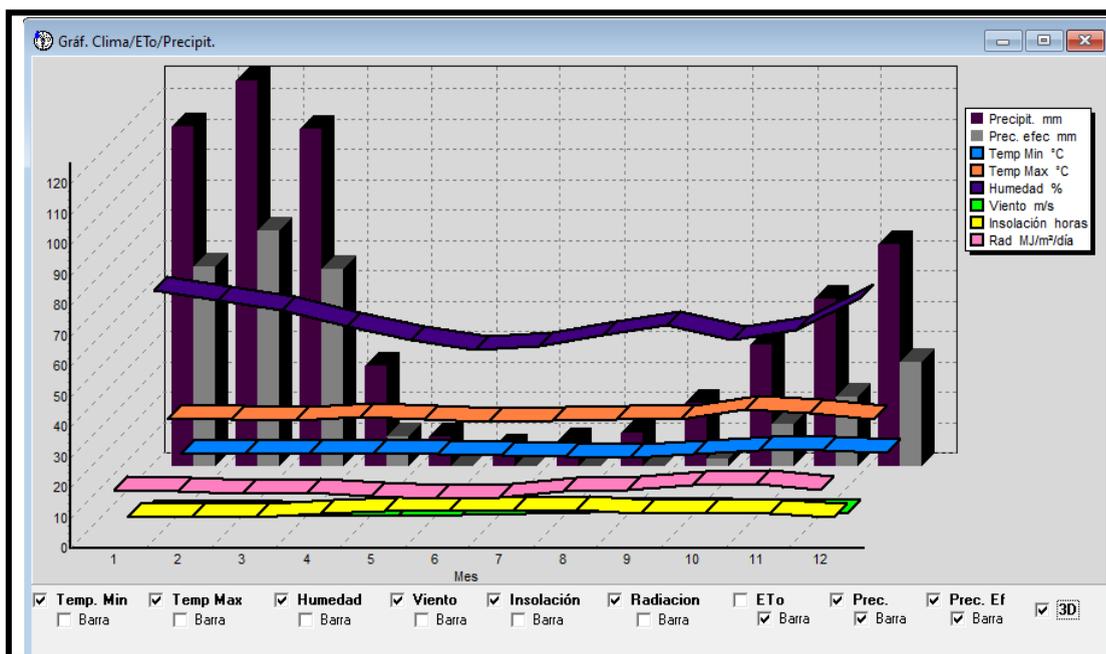
FUENTE: CROPWAT 8.0

Figura 11 — Importación de precipitación



FUENTE: CROPWAT 8.0

Figura 12 — Histograma de las precipitaciones de la ciudad de Abancay



Valores del coeficiente único (promedio temporal) del cultivo,  $K_c$ , y alturas medias máximas de las plantas para cultivos no estresados y bien manejados en climas sub-húmedos ( $HR_{min} = 45\%$ ,  $u_2 = 2 \text{ m s}^{-1}$ ) para usar en la fórmula de la FAO Penman-Monteith  $ET_c$ .

Cultivo	$K_{c,med}$	$K_{c,max}$	$K_{c,min}$	Altura Máx. Cultivo (h) (m)	
a. Hortalizas Pequeñas	0,7	1,05	0,95		
Brécol (Brócoli)		1,05	0,95	0,3	
Col de Bruselas		1,05	0,95	0,4	
Repollo		1,05	0,95	0,4	
Zanahoria		1,05	0,95	0,3	
Coliflor		1,05	0,95	0,4	
Apio (Céleri)		1,05	1,00	0,6	
Ajo		1,00	0,70	0,3	
Lechuga		1,00	0,95	0,3	
Cebolla - seca		1,05	0,75	0,4	
- verde		1,00	1,00	0,3	
- semilla		1,05	0,80	0,5	
Espinaca		1,00	0,95	0,3	
Rábano		0,90	0,85	0,3	
b. Hortalizas- Familia de la Solanáceas	0,6	1,15	0,80		
Berenjena		1,05	0,90	0,8	
Pimiento Dulce (campana)		1,05 <sup>2</sup>	0,90	0,7	
Tomate		1,15 <sup>2</sup>	0,70-0,90	0,6	
c. Hortalizas- Familia de las Cucurbitáceas	0,5	1,00	0,80		
Melón		0,5	0,85	0,60	0,3
Pepino - Cosechado Fresco		0,6	1,00 <sup>2</sup>	0,75	0,3
- Cosechado a Máquina		0,5	1,00	0,90	0,3
Calabaza de Invierno		1,00	0,80	0,4	
Calabacín (zucchini)		0,95	0,75	0,3	
Melón dulce		1,05	0,75	0,4	
Sandía		0,4	1,00	0,75	0,4
d. Raíces y Tubérculos	0,5	1,10	0,95		
Remolacha, mesa		1,05	0,95	0,4	
Yuca o Mandioca - año 1		0,3	0,80 <sup>1</sup>	0,30	1,0
- año 2		0,3	1,10	0,50	1,5

FUENTE: CROPWAT 8.0

Figura 13 — Documento de la FAO para el cálculo de la evapotranspiración



### C. Duración de las etapas de crecimiento del cultivo para distintos períodos de siembra y regiones climáticas (días)

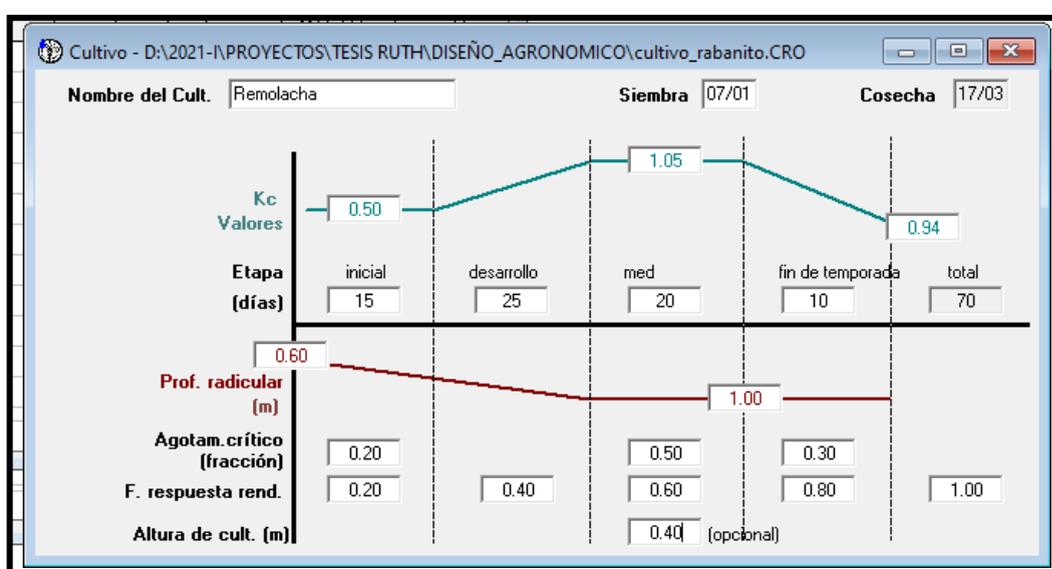
CUADRO 11 (continuación)

Cultivo	Inic. (L <sub>in</sub> )	Des. (L <sub>des</sub> )	Med. (L <sub>med</sub> )	Final (L <sub>fin</sub> )	Total	Fecha de Siembra	Región
Melón Dulce	25	35	40	20	120	Mayo	Mediterráneo
	30	30	50	30	140	Marzo	Calif., EU
	15	40	65	15	135	Ago	Calif. Desierto, EU
	30	45	65	20	160	Dic/Ene.	Región Árida
Sandía	20	30	30	30	110	Abril	Italia
	10	20	20	30	80	May/Ago.	Cercano Este (desierto)
<b>d. Raíces y Tubérculos</b>							
Remolacha, mesa	15	25	20	10	70	Abr/Mayo	Mediterráneo
	25	30	25	10	90	Feb/Mar.	Mediterráneo & Árido
Mandioca o Yuca: año 1	20	40	90	60	210	Período Lluvioso	Regiones Tropicales
	150	40	110	60	360		
Patata o Papa	25	30	30/45	30	115/130	Ene/Nov.	Clima (Semi)Árido
	25	30	45	30	130	Mayo	Clima continental
	30	35	50	30	145	Abril	Europa
	45	30	70	20	165	Abr/Mayo	Idaho, EU
	30	35	50	25	140	Dic.	Calif. Desierto, EU
Camote o Batata	20	30	60	40	150	Abril	Mediterráneo
	15	30	50	30	125	Período Lluvioso	Regiones Tropicales
Remolacha azucarera	30	45	90	15	180	Marzo	Calif., EU
	25	30	90	10	155	Junio	Calif., EU
	25	65	100	65	255	Sept.	Calif. Desierto, EU
	50	40	50	40	180	Abril	Idaho, EU

FUENTE: FAO

Figura 14 — Duración de las etapas de crecimiento del cultivo

Según el manual de la FAO, el tiempo de siembra recomendado para el cultivo de remolacha es de 70 días. En las figuras 16 y 17 se muestra el valor del requerimiento de agua para este cultivo.



FUENTE: CROPWAT 8.0

Figura 15 — Diseño de cultivo remolacha



Suelo - D:\2021-1\PROYECTOS\TESIS RUTH\DISEÑO\_AGRONOMICO\diseño\_suelo.SOI

Nombre del suelo: Franco Arenoso

Datos generales de suelo:

- Humedad de suelo disponible total (CC-PMP): 150.0 mm/metro
- Tasa máxima de infiltración de la precipitación: 2 mm/día
- Profundidad radicular máxima: 50 centímetros
- Agotamiento inicial de hum. de suelo (como % de ADT): 0 %
- Humedad de suelo inicialmente disponible: 150.0 mm/metro

FUENTE: CROPWAT 8.0

Figura 16 — Requerimiento de agua para el cultivo de remolacha

Requerimiento de Agua del Cultivo

Estación ET: ABANCAY      Cultivo: Remolacha

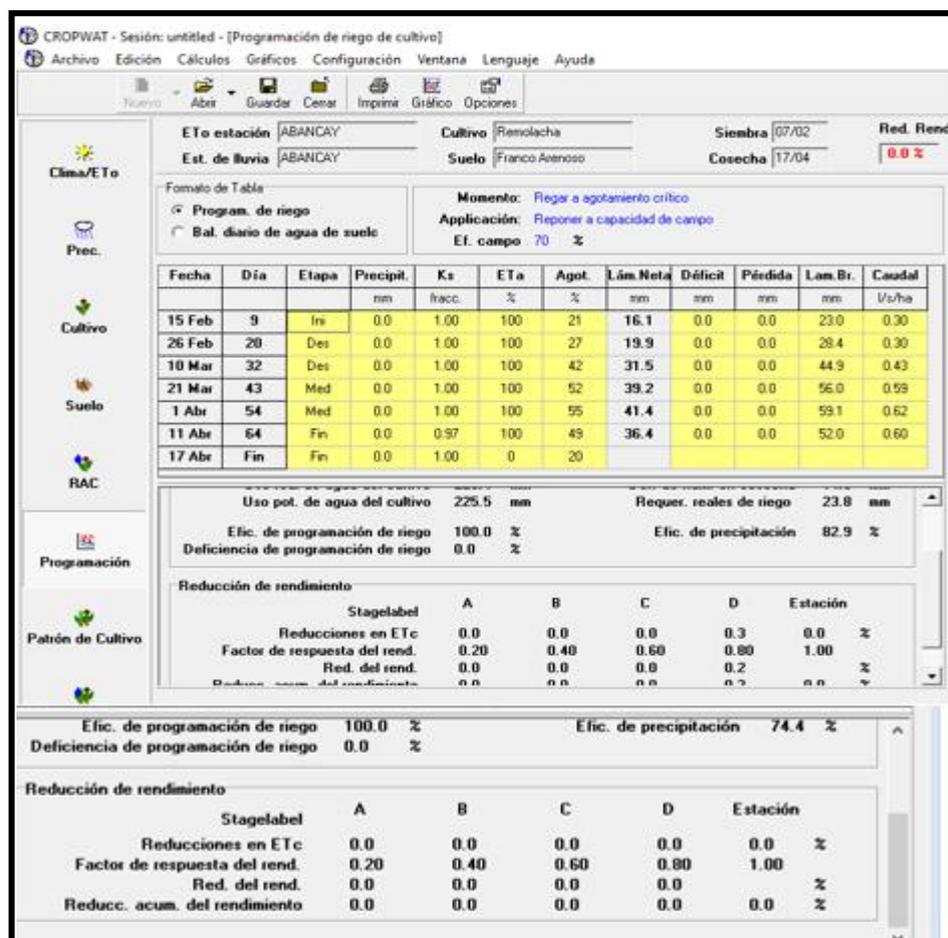
Est. de lluvia: ABANCAY      Fecha de siembra: 07/02

Mes	Decada	Etap	Kc	ETc	ETc	Prec. efec	Req. Riego
			coef	mm/día	mm/dec	mm/dec	mm/dec
Feb	1	Inic	0.50	2.02	8.1	10.1	0.0
Feb	2	Inic	0.50	2.01	20.1	27.0	0.0
Feb	3	Des	0.58	2.31	18.5	25.2	0.0
Mar	1	Des	0.78	3.08	30.8	24.3	6.5
Mar	2	Med	1.00	3.91	39.1	23.6	15.5
Mar	3	Med	1.06	4.13	45.4	16.8	28.6
Abr	1	Fin	1.06	4.08	40.8	8.0	32.9
Abr	2	Fin	0.99	3.79	26.5	0.8	25.3
					229.3	135.8	108.9

FUENTE: CROPWAT 8.0

Figura 17—Requerimiento de agua para el cultivo de remolacha

### D. Interfaz de la programación del cultivo de remolacha

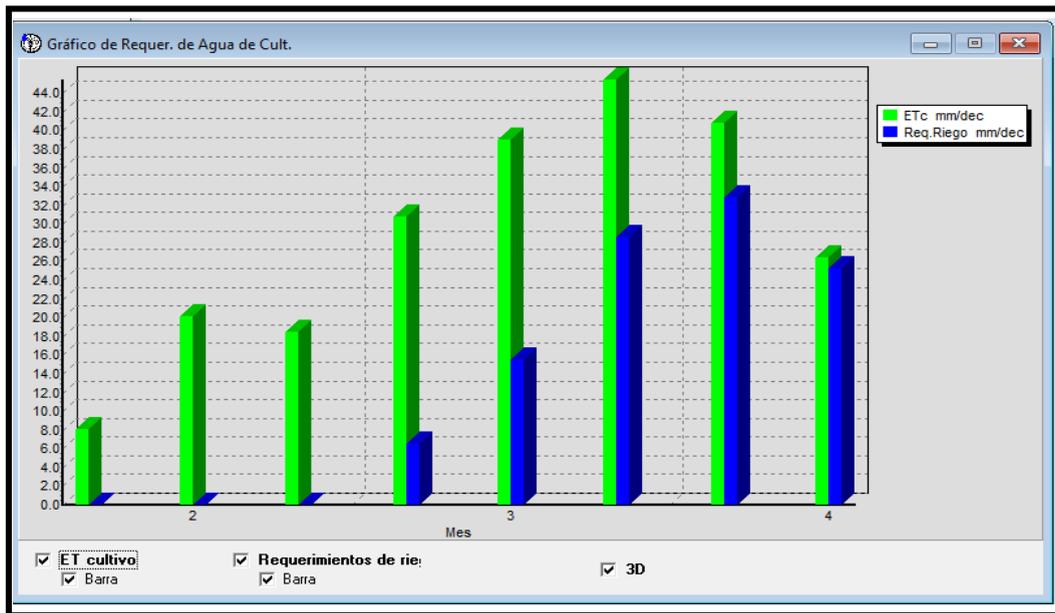


FUENTE: CROPWAT 8.0

Figura 18 — Interfaz de programación del cultivo

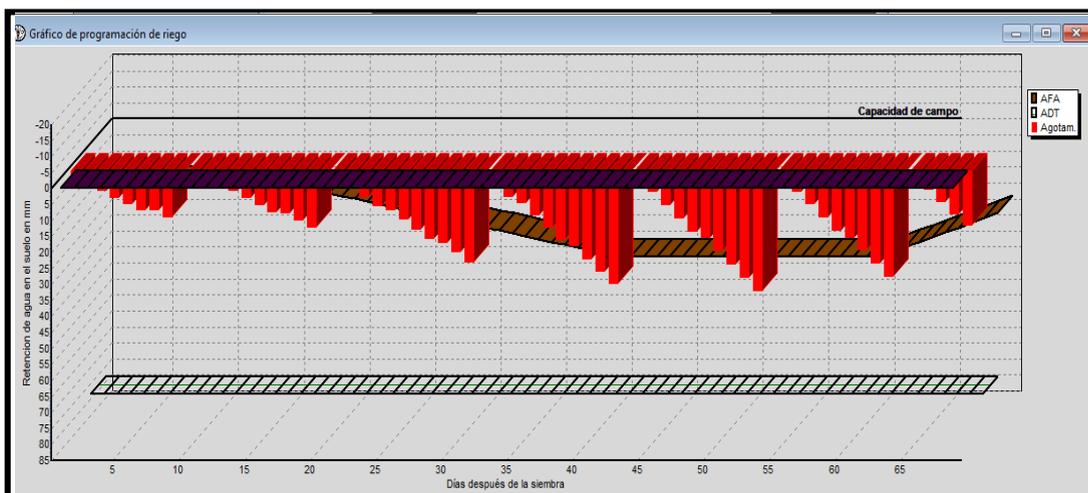
En la Figura 18, 19 y 20 se muestra la interfaz de programación del cultivo, el requerimiento del agua y la programación del cultivo en sí.





FUENTE: CROPWAT 8.0

Figura 19 — Requerimiento de agua para el cultivo, y la ET0



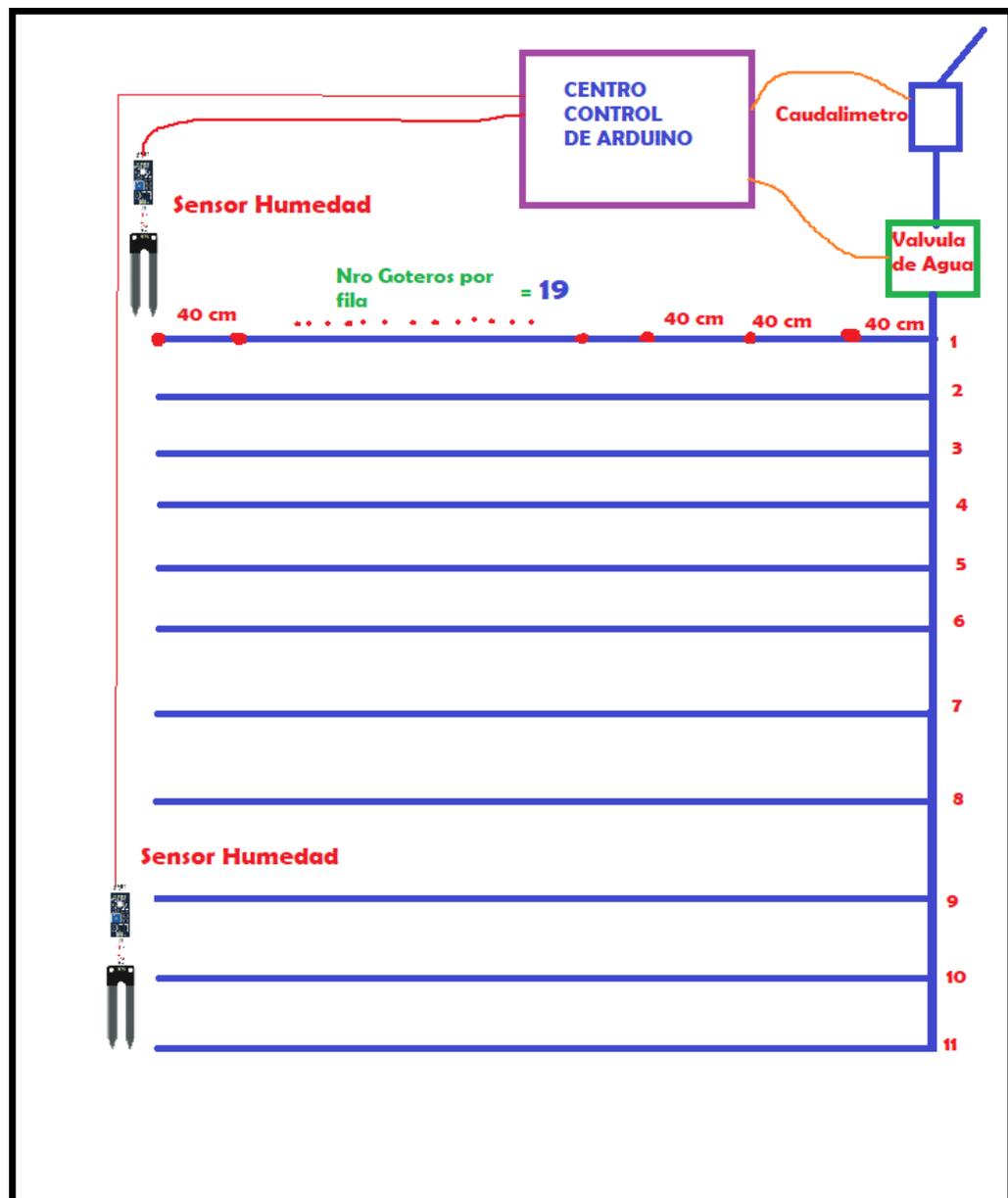
FUENTE: CROPWAT 8.0

Figura 20 — Programación de cultivo

### **5.1.2 Diseño hidráulico para el riego tecnificado por goteo**

El diseño hidráulico es una parte fundamental en el diseño de sistemas de riego tecnificado, ya sea por goteo, aspersión u otros tipos de diseño. En este estudio, se utilizó el diseño de riego por goteo, que se destaca por ser altamente eficiente en términos de manejo de tiempos de riego y control preciso de la cantidad de agua suministrada en un periodo determinado. El diseño hidráulico implica la configuración de tuberías de manera específica, con una tubería principal y subunidades o ramales donde se colocan los goteros individuales en cada planta, según la subdivisión de la parcela. A continuación, se presentará el diseño o estructura de la tubería de riego junto con los materiales correspondientes.

### A. Esquema de diseño hidráulico



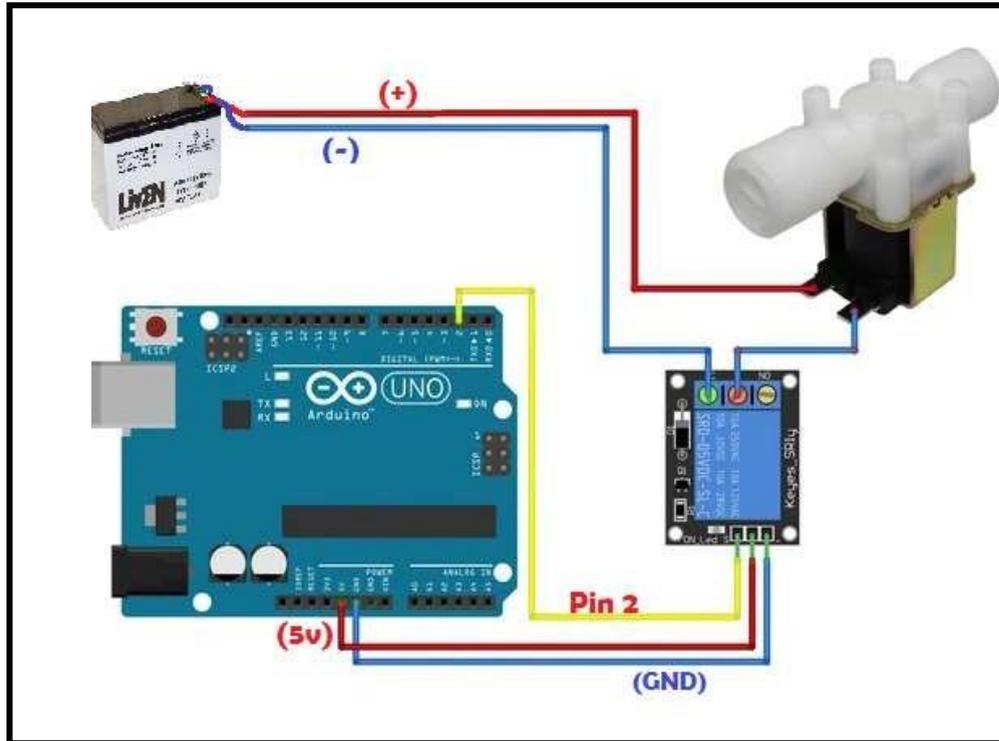
FUENTE: elaboración propia

**Figura 21 — Diseño hidráulico de riego por goteo**

### B. Diseño de la válvula en el Arduino

En el siguiente esquema se muestra el esquema del circuito de la válvula de agua, relay, batería, y Arduino todos estos componentes se conectan para controlar la cantidad de agua que debe pasar por la tubería y el tiempo de necesario para el riego.



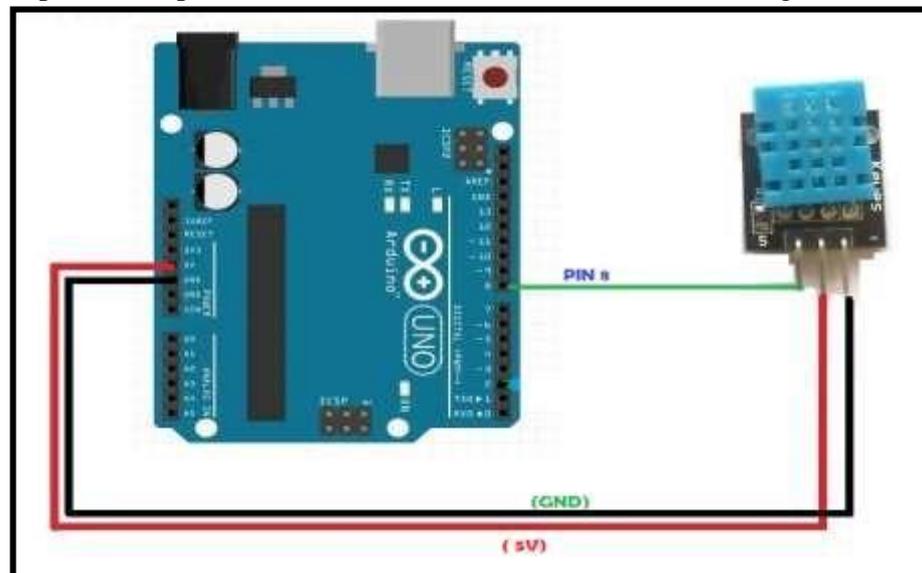


FUENTE: elaboración propia

Figura 22 — Diseño de válvula

### C. Diseño de DHT – Sensor de Temperatura en Arduino

El sensor DHT es un tipo de sensor que obtiene la temperatura ambiente, es un parámetro para tomar en cuenta a la hora de realizar el riego.

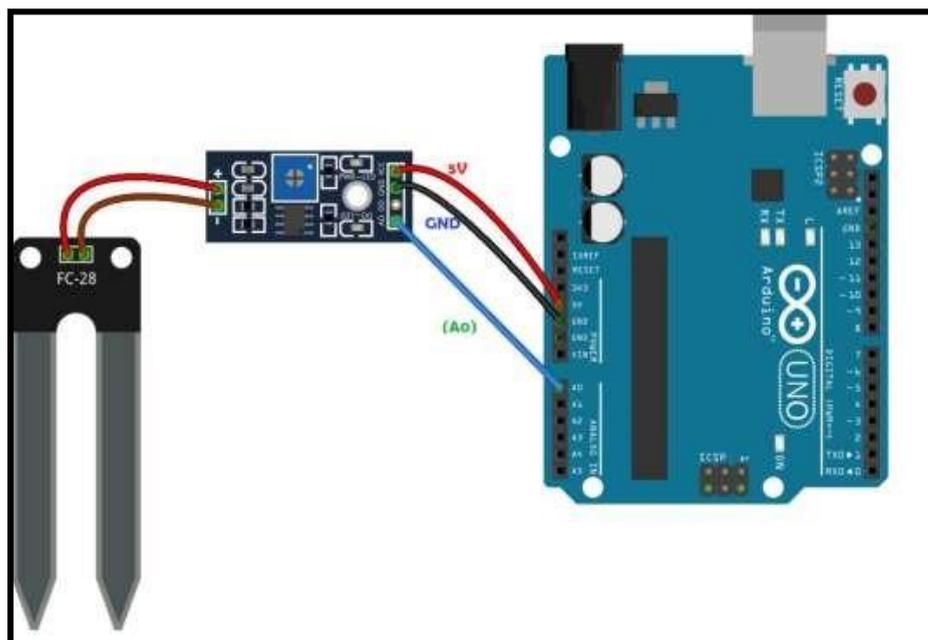


FUENTE: elaboración propia

Figura 23 — Diseño de DHT

#### D. Esquema de sensor de humedad

El sensor de humedad nos ayudara a calcular hasta que profundidad llego el agua y como es la concentración la cantidad de humedad tiene el suelo mientras la válvula este abierta.



FUENTE: elaboración propia

**Figura 24** — Esquema sensor de humedad

#### 5.1.3 Programación en Arduino UNO para el funcionamiento del riego por goteo

Se programó en la plataforma Arduino para el control del recurso hídrico.

```
#define RELE 2 #include "DHT.h"
// #define DHTTYPE DHT11 // DHT 11
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
const int DHTPin = 8; // what digital pin we're connected to DHT
dht(DHTPin, DHTTYPE);
const int TIEMPO_RIEGO = 60000; const int TIEMPO_ESPERA = 10000;
// LECTURA Y CALCULO DE CAUDALIMETRO
volatile int NumPulsos; // variable para la cantidad de pulsos recibidos int
```

```
PinSensor = 1;           //Sensor conectado en el pin 2
float factor_conversion=7.5; //para convertir de frecuencia a caudal
//---Función que se ejecuta en interrupción-----
void ContarPulsos ()
{
  NumPulsos++; //incrementamos la variable de pulsos
}
//---Función para obtener frecuencia de los pulsos-----
int ObtenerFrecuencia(){
  int frecuencia;
  NumPulsos = 0; //Ponemos a 0 el número de pulsosinterrupts();
                //Habilitamos las interrupciones delay(1000); //muestra
                de 1 segundo noInterrupts(); //Desabilitamos las interrupciones
  frecuencia=NumPulsos; //Hz(pulsos por segundo) return frecuencia;
}
//TERMINO DE CAUDALIMETRO
void setup() {
  //Paso 1 Serial.begin(9600);
  Serial.println("Valor del sensor de humedad");
  //RELE PARA ELECTROVALVULA
  pinMode(RELE, OUTPUT); //Define el pin RELE como salida
  //lectura de CAUDALIMETROpinMode(PinSensor, INPUT);
  attachInterrupt(0, ContarPulsos, RISING); //(Interrupcion
0(Pin2),funcion,Flanco desubida)
}
void loop() {
  //SENSOR DE TEMPERATURA
  // Reading temperature or humidity takes about 250 milliseconds!float h =
  dht.readHumidity();
  float t = dht.readTemperature();
  //Serial.print("\n HUMEDAD: ");
  //Serial.print(h); Serial.println("\n"); Serial.print("TEMPERATURA:");t=18;
  Serial.print(t); Serial.print(" *C ");Serial.println("\n");
```



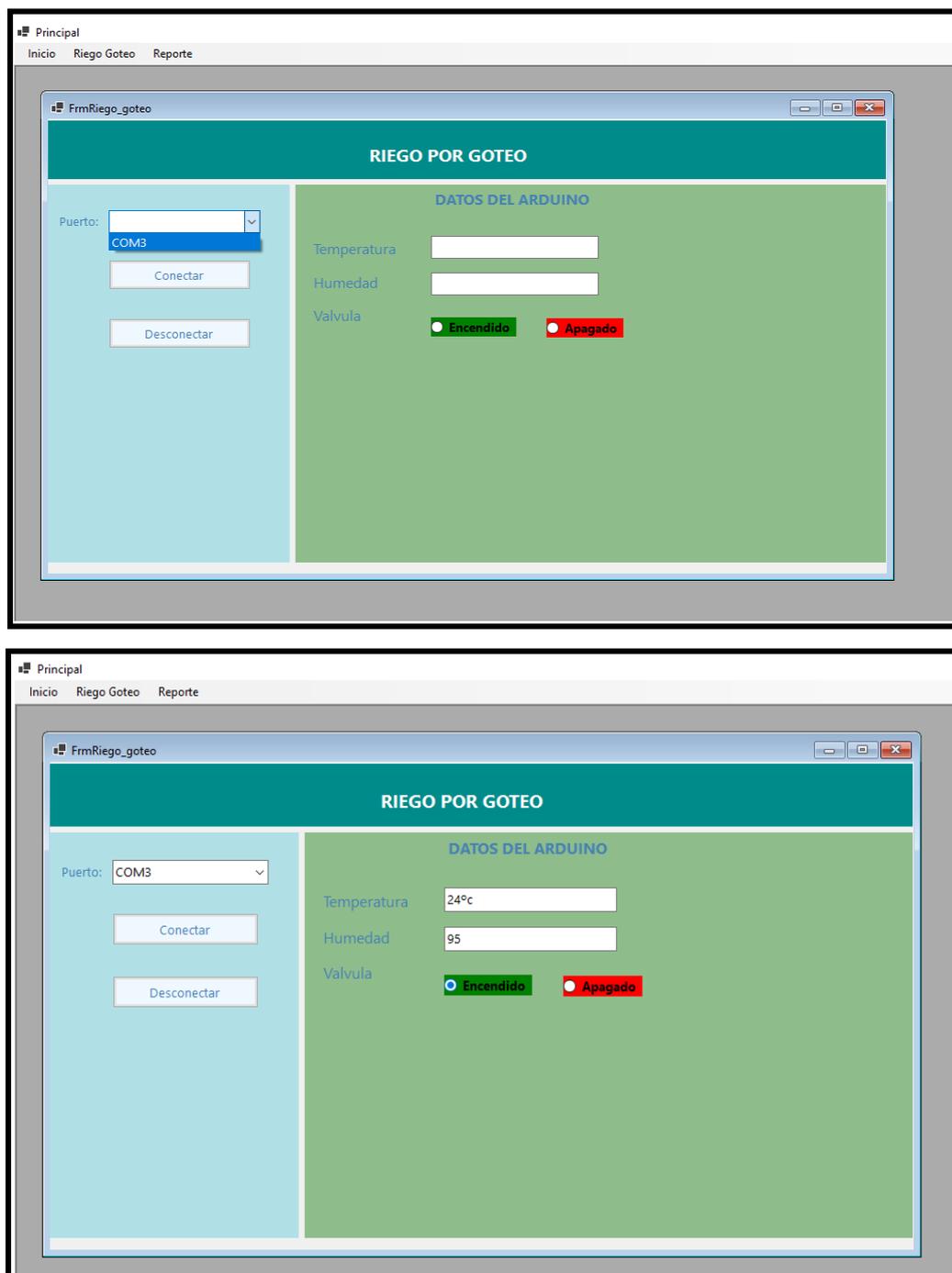
**//Paso 2**

```
int humidity = analogRead(A0); int humidity2 = analogRead(A2);
Serial.print("\nLectura sensor 1:");Serial.print(humidity); Serial.println("");
Serial.print("Lectura sensor 2:"); Serial.print(humidity2); Serial.println("");
//lectura de caudalimetro
float frecuencia=ObtenerFrecuencia(); //obtenemos la Frecuencia de los pulsos enHz
float caudal_L_m=frecuencia/factor_conversion; //calculamos el caudal en L/m
float caudal_L_h=caudal_L_m*60; //calculamos el caudal en L/h
//-----Enviamos por el puerto serie-----
Serial.println("FrecuenciaPulsos: ");Serial.print(frecuencia,0); Serial.print(" Hz\tCaudal: ");
caudal_L_m=0.24; Serial.print(caudal_L_m,3); Serial.print(" L/m\t");
Serial.print(caudal_L_h,3); Serial.print(" L/h"); Serial.print("\n");
//termino de caudalimetro
//Paso 3
if(humidity >= 0 &( humidity <= 700 && humidity2<=700)){
    Serial.println("\n Sensor en suelo humedo"); digitalWrite(RELE, LOW);//APAGA el rele
    delay(1000);
    Serial.println("\n :: VALVULA APAGADA ::\n");
    //delay(TIEMPO_RIEGO);//Espera el tiempo de llenado
}else if((humidity >= 700) &&(humidity2 >= 700)){Serial.println("\n Sensor en suelo seco");
    digitalWrite(RELE, HIGH);//VALVULA PRENDIDA el reldelay(10000);
    Serial.println("\n :: VALVULA PRENDIDA ::\n");
    // delay(TIEMPO_DE_LLENADO);//Tiempo de espera antes de repetir el proceso
}
delay(1000);
}
```



### 5.1.4 Interfaz del software con tecnología arduino

En la figura 25, se visualiza la interfaz del software.



FUENTE: elaboración propia

Figura 25 – Interfaz gráfica de usuario del software con tecnología arduino

### 5.1.5 Resultado del diseño hidráulico

En la figura 26,27,28 y 29 se muestra el diseño hidráulico de las instalaciones de riego por goteo.



FUENTE: elaboración propia

**Figura 26** —Ramal a ramal 40cm de un gotero a otro



FUENTE: elaboración propia

**Figura 27** —Líneas terciarias instaladas con la tecnología de riego por goteo



FUENTE: elaboración propia

**Figura 28** —Goteros en cada subunidad



FUENTE: elaboración propia

**Figura 29** — Sensor de humedad instalado en la raíz de una remolacha

#### **5.1.6 Análisis de resultados de la investigación de la calidad de la eficiencia de desempeño del software con tecnología arduino.**

Los resultados de la Tabla 2 muestran el puntaje final de la calidad de eficiencia de desempeño, donde se le asigna una importancia “A” porque indica la necesidad de realizar mediciones para evaluar el rendimiento del sistema considerando los recursos que se utilizan. Se le otorga una ponderación del 100% debido a la necesidad de evaluar el rendimiento del sistema teniendo en cuenta los recursos que se utilizaron.

**Tabla 2 — Resultado de medir la eficiencia de desempeño del software con tecnología arduino**

Subcaracterística	Métrica	Fórmula	Valores deseados (especificar valores)	Aplica	Valor obtenido (X)	Ponderación (/10)	Valor total parcial (/10)	Nivel de Importancia	% de la importancia	Valor Final
Comportamiento Tiempo	Tiempo de respuesta	$X=B-A$ A=Tiempo de envío de petición B=Tiempo en recibir la primera respuesta	Deseado: 14seg Peor caso>14 seg	SI	A=0.253 B=9.973 X=9.72	97.2	58.70	A	100%	6.87
	Tiempo de espera	$X=B-A$ A=Tiempo cuando se inicia el trabajo B=Tiempo en completar el trabajo	Deseado: 20seg Peor caso>20 seg	SI	A=0.353 B=11.909 X=11.556	115.56				
	Rendimiento	$X=-A/T$ A=Número de tareas completadas T=Intervalo de tiempo (donde $T>0$ )	Deseado: 7/11min Peor caso:0/11min	SI	A= 10  T= 4.818 X= 2.075	20.75				
Utilización de recursos	Utilización de CPU	$X=(A/100) *100\%$ A=Cantidad de CPU que es usado por una tarea	Deseado: 1% Peor caso $\geq 50\%$	SI	A=1 X=1	10				
	Utilización de la memoria	$X=A/B$ A=Cantidad de memoria que es usado para realizar una tarea	Deseado: 1% Peor caso $\geq 10\%$	SI	A=5 X=5	50				
	Utilización de los dispositivos de E/S	$X=B-A$ A=Tiempo que los dispositivos de E/S pasan ocupados en realizar una tarea B=Tiempo de operación donde $B>0$	Deseado $\geq 10/3$ min Pero caso $0/3$ min	NA	NA	NA				

## 5.2 Contrastación de hipótesis

### 5.2.1 Contrastación hipótesis para la eficiencia de riego

#### Hipótesis general

La automatización de riego tecnificado por goteo mejora la eficiencia de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

En la Tabla 2, se visualiza los parámetros del tiempo de respuesta con una ponderación de 97.2, tiempo de espera con una ponderación de 115.56 y el rendimiento con una ponderación de 20.75 por lo que se tiene el valor final con 6.87 puntos, considerando los niveles de puntuación para el grado de satisfacción de la eficiencia del desempeño del software, se puede concluir que la automatización del riego tecnificado por goteo mediante el uso del software con tecnología Arduino tiene un nivel de puntuación "ACEPTABLE" y es satisfactorio.

### 5.2.2 Contrastación hipótesis para el tiempo de riego

#### Hipótesis específica 1

La automatización de riego tecnificado por goteo mejora el tiempo de riego mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

#### a) Datos estadísticos

$N = n = 11$  filas con 19 goteros en cada fila, se analizará solo en los últimos goteros de cada fila.

Tiempo de llegada a los últimos goteros de las 11 filas

Tabla 3 — Datos estadísticos

Nro . de fila	Tiempo de llegada del agua al último gotero con el sistema (Seg.)	tiempo de llegada del agua al último gotero sin el sistema (Seg.)
1	33	35
2	35	45
3	40	46
4	45	48
5	50	55
6	55	60
7	60	67
8	65	69
9	70	71
10	75	76
11	80	81

FUENTE: elaboración propia

**b) Hipótesis estadística**

Ho:  $\mu A = \mu B$  [El tiempo de riego con la automatización de riego tecnificado por goteo mediante el uso del software es igual sin el desarrollo e implementación de un software con tecnología Arduino, 2021].

Ha:  $\mu A \neq \mu B$  [El tiempo de riego con la automatización de riego tecnificado por goteo mediante el uso del software es diferente sin el desarrollo e implementación de un software con tecnología Arduino, 2021].

**c) Nivel de Significancia  $\alpha = 5 \% = 0,05$**

**d) Estadístico**

Como las muestras son pequeñas,  $n=m=11$ ,  $n, m < 30$  entonces se usa la distribución t-student con “ $n+m-2$ ” grados de libertad.

N: tamaño de la población, m: muestras

$n=m=11$



- *Fórmula para el cálculo de la media*

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

- *Desviación estándar*

$$\sqrt{\frac{\sum_{i=1}^n x_i^2}{N} - (\bar{x})^2} \quad \text{o} \quad S = \delta = \sqrt{\frac{\sum_{i=1}^n (x_i^2 - n\bar{x}^2)}{n-1}}$$

- *T- Calculado*

$$T_c = \frac{\bar{x}_A - \bar{x}_B}{\sqrt{\frac{\delta_c^2}{n_A} + \frac{\delta_c^2}{n_B}}}$$

- *Calcularemos la varianza en común*

$$\delta_c = \frac{(n_A - 1)\delta_A^2 + (n_B - 1)\delta_B^2}{n_A + n_B - 2}$$

$$\bar{x}_A = 55.27$$

$$\delta_A^2 = 260.82$$

$$\delta_A = 16.15$$

$$\bar{x}_B = 59.3636$$

$$\delta_B^2 = 215.855$$

$$\delta_B = 14.69199$$

$$\delta_c = \frac{(11 - 1)(260.82) + (11 - 1)(215.855)}{11 + 11 - 2}$$

$$\delta_c = 238.3375$$

$$T_c = \frac{55.27 - 59.3636}{\sqrt{\frac{238.82}{11} + \frac{238.82}{11}}}$$

$$T_c = -0.6212285$$



e) **Región crítica**

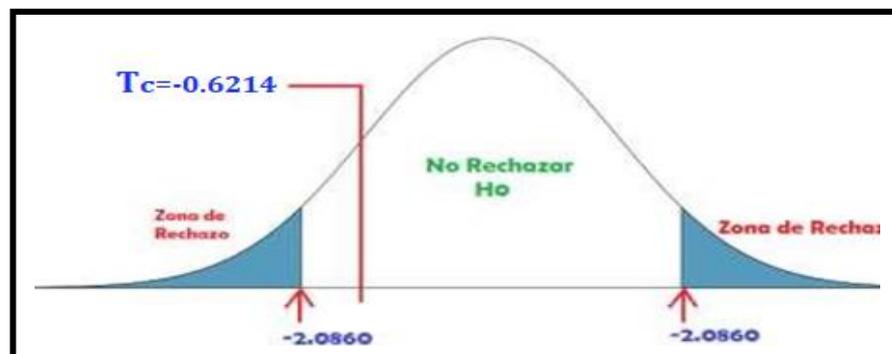
$$p(T_c - T_0) = \frac{a}{2}, \text{ con } (n + m - 2) g - l$$

$$1 - p(T_c > T_0) = \frac{a}{2}$$

$$1 - \frac{a}{2} = p(T_c < T_0) \text{ con } (m + n - 2)g - l$$

$$1 - 0.05 = p(T_c < T_0) \text{ con } 11 + 11 - 2 = 20 g - l$$

$$0.975 = p(T_c < 2.0860) \text{ con } 20 g - l$$

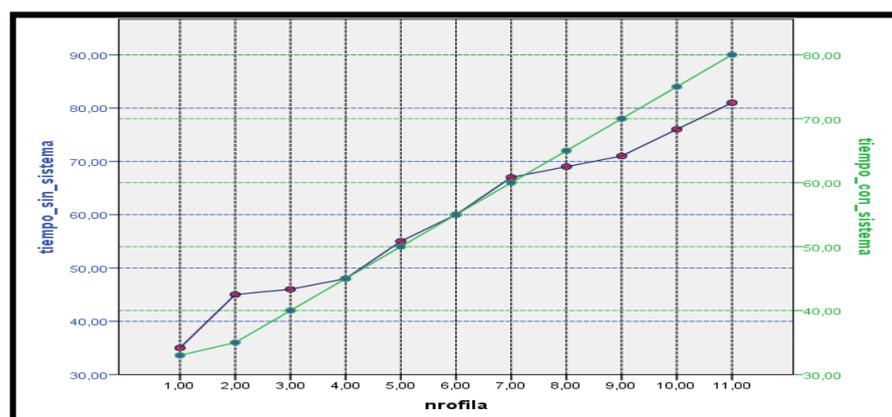


FUENTE: elaboración propia

**Figura 30** — Curva de región crítica de la hipótesis general

f) **Interpretación**

En la figura 31, se muestra el gráfico de ejes dobles, para mostrar la distribución de los datos aplicando el software y sin aplicar el software con tecnología Arduino.



FUENTE: MINITAB

**Figura 31** — Gráfico de tiempo de riego con la aplicación y sin la aplicación

Como:  $T_c = -0.6212285 \in < -\infty; -2.0860 > U < +2.0860; +\infty >$ . Por lo tanto, se acepta la hipótesis  $H_a$ , se concluye que al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino es diferente al tiempo de riego sin aplicar el software.

### 5.2.3 Contrastación de hipótesis para el consumo de agua

#### Hipótesis específica 2

La automatización de riego tecnificado por goteo mejora el consumo de agua mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

#### a) Datos estadísticos

$N = n = 11$  filas con 19 goteros en cada fila, se analizará solo en los últimos goteros de cada fila.

Cantidad de agua que debe llegar en el último gotero por un determinado periodo.

**Tabla 4 — Datos estadísticos con sistema y sin sistema**

Nro Fila	Con sistema Cantidad de agua (Lts/seg)	Sin sistema Cantidad de agua (Lts/seg)
1	0.24	0.32
2	0.27	0.36
3	0.30	0.30
4	0.24	0.24
5	0.32	0.40
6	0.38	0.45
7	0.39	0.39
8	0.40	0.50
9	0.42	0.64
10	0.44	0.70
11	0.45	0.75

FUENTE: elaboración propia

#### *Hipótesis estadística*

$H_0: \mu_A > \mu_B$  [El consumo de agua con la automatización de riego tecnificado



por goteo mediante el uso del software es mayor al consumo del agua sin el desarrollo e implementación de un software con tecnología Arduino, 2021].

Ha:  $\mu A < \mu B$  [El consumo de agua con la automatización de riego tecnificado por goteo mediante el uso del software es menor al consumo del agua sin el desarrollo e implementación de un software con tecnología Arduino, 2021].

b) **Nivel de Significancia**  $\alpha = 5 \% = 0,05$

c) **Estadístico**

Como las muestras son pequeñas,  $n=m=11$ ,  $n, m < 30$  entonces se usa la distribución t-student con “ $n+m-2$ ” grados de libertad.

N: tamaño de la población, m: muestras

**$n=m=11$**

$$\bar{x}_A = 0.35$$

$$\delta_A^2 = 0.0787$$

$$\delta_A = 0.0062$$

$$\bar{x}_B = 0.459090$$

$$\delta_B^2 = 0.1697$$

$$\delta_B = 0.0288$$

$$\delta_c = \frac{(11 - 1)(0.0787) + (11 - 1)(0.1697)}{11 + 11 - 2}$$

$$\delta_c = 0.017495$$

$$T_c = \frac{0.35 - 0.4591}{\sqrt{\frac{0.017495}{11} + \frac{0.017495}{11}}}$$

$$T_c = -1.9344$$

$$p(T_c - T_0) = \frac{\alpha}{2}, \text{ con } (n + m - 2) g - l$$

$$1 - p(T_c > T_0) = \frac{\alpha}{2}$$

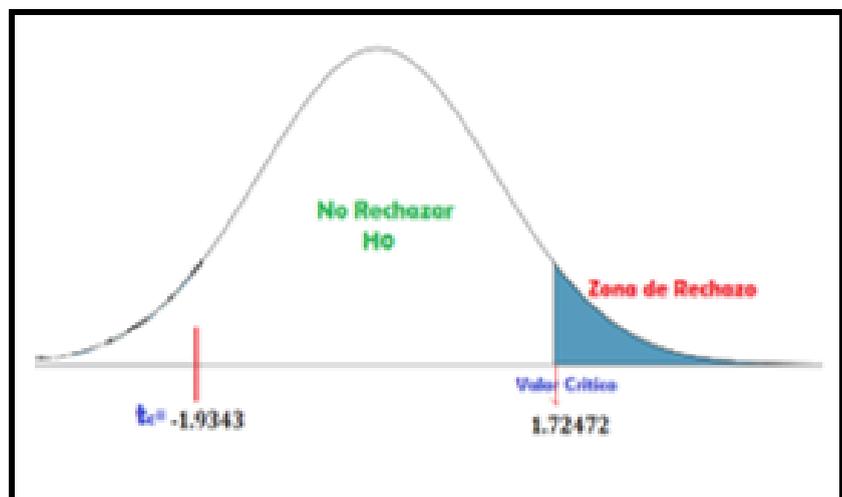
$$1 - \frac{\alpha}{2} = p(T_c < T_0) \text{ con } (m + n - 2)g - l$$

$$1 - 0.05 = p(T_c < T_0) \text{ con } 11 + 11 - 2 = 20 g - l$$

$$0.975 = p(T_c < 2.0860) \text{ con } 20 g - l$$

**d) Región crítica**

$$0.975 = p(tc < 1.7247) \text{ con } 20 g - l$$

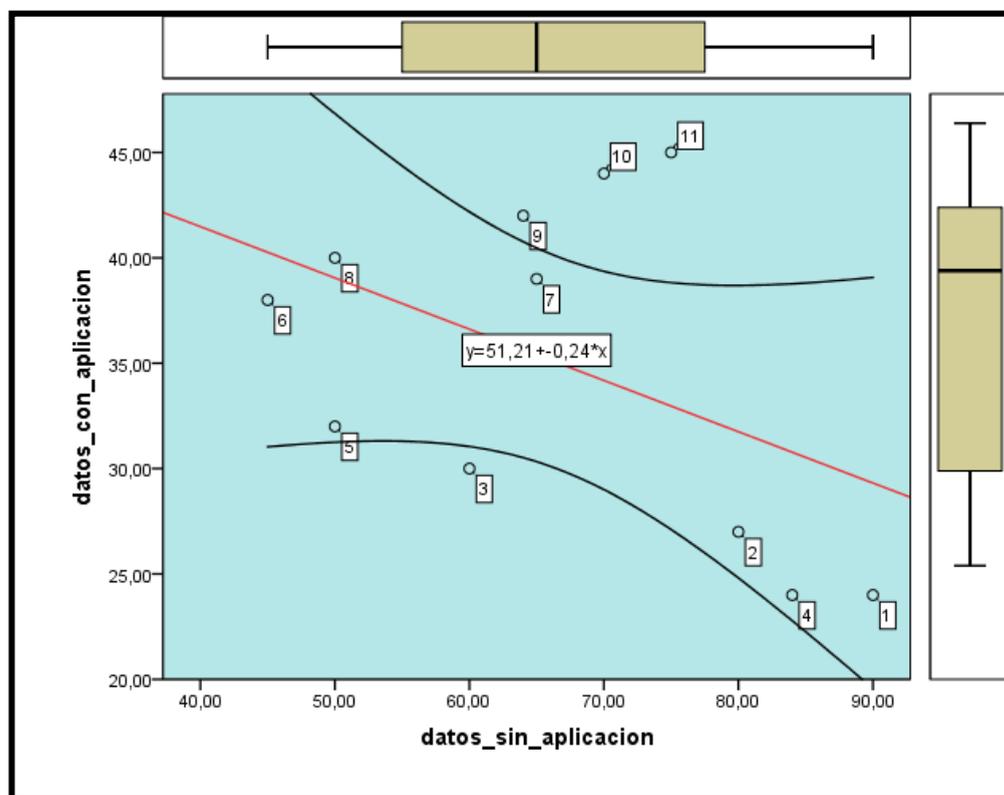


FUENTE: elaboración propia

**Figura 32 — Gráfico de región crítica**

e) **Interpretación**

En la figura 33 se muestra la diferencia que existe entre un grupo de datos con respecto al otro indicando que los datos de pérdida de agua en el primer grupo que es la cantidad de agua que se abaste con la aplicación es menor que la cantidad de agua que se abastece sin la aplicación. Esto nos indica una mayor pérdida de agua sin aplicación y mayor control con el software con tecnología Arduino.



FUENTE: Minitab v5

**Figura 33 — Gráfico de pérdida de agua con y sin el sistema**

Como  $tc = -1.9344 \in < -\infty; +1.7247 >$  entonces se acepta la hipótesis  $H_a$ , esto confirma que al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, se logra mejorar el consumo de agua. Esto se debe a que el consumo de agua es menor en comparación con la ausencia de la aplicación del software de riego tecnificado por goteo.



### 5.2.4 Contrastación de hipótesis para el nivel de humedad

#### Hipótesis específica 3

La automatización de riego tecnificado por goteo mejora el nivel de humedad del suelo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021.

#### a) Datos estadísticos

$N = n = 11$  filas con 19 goteros en cada fila, se analizará solo en los últimos goteros de cada fila.

Control del porcentaje de humedad que debe tener en el último gotero en cada fila por un determinado periodo.

**Tabla 5 —Datos estadísticos con sistema y sin sistema**

Nro. Fila	porcentaje Humedad (%) Sin el sistema (A)	porcentaje Humedad (%) Con sistema (B)	Diferencia
1	1010.5	953	57.5
2	1012	988	24
3	1020	987.7	32.3
4	989	975	14
5	1010.5	852	158.5
6	1005	800	205
7	989	750	239
8	963.3	768	195.3
9	890	874	16
10	978.6	963	15.6
11	1030.1	875	155.1

FUENTE: elaboración propia

#### b) Hipótesis estadística

$H_0: \mu_A \leq \mu_B$  [La automatización de riego tecnificado por goteo no mejora el nivel de humedad del suelo mediante el desarrollo e implementación de un



software con tecnología Arduino, 2021].

Ha:  $\mu_A > \mu_B$  [La automatización de riego tecnificado por goteo mejora el nivel de humedad del suelo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021].

c) **Nivel de Significancia**  $\alpha = 5 \% = 0,05$

d) **Estadístico**

Como las muestras son pequeñas,  $n=m=11$ ,  $n, m < 30$  entonces se usa la distribución t-student con “ $n+m-2$ ” grados de libertad.

N: tamaño de la población, m: muestras

$n=m=11$

$$t_c = \frac{\bar{d}}{s_d / \sqrt{n}}$$

*Donde:*

$\bar{d}$ : es la media de la diferencia de los valores de humedad

$$\bar{d} = 101118$$

$$s_d = 892263$$

$$n = 11$$

$$t_c = 3.75865$$

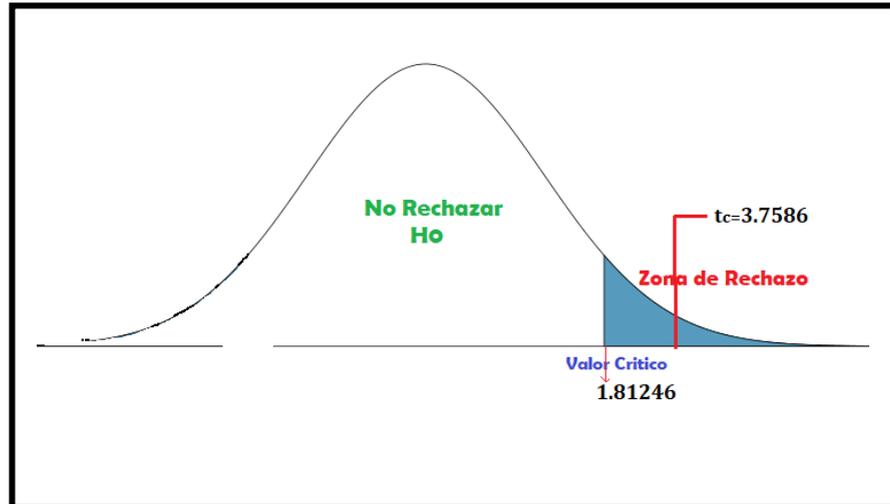
e) **Región crítica**

$$gl = n - 1 = 11 - 1 = 10$$

$$0.975 = p(tc < 1.8127) \text{ con } 10 \text{ g} - l$$

$$\alpha = 0.05$$

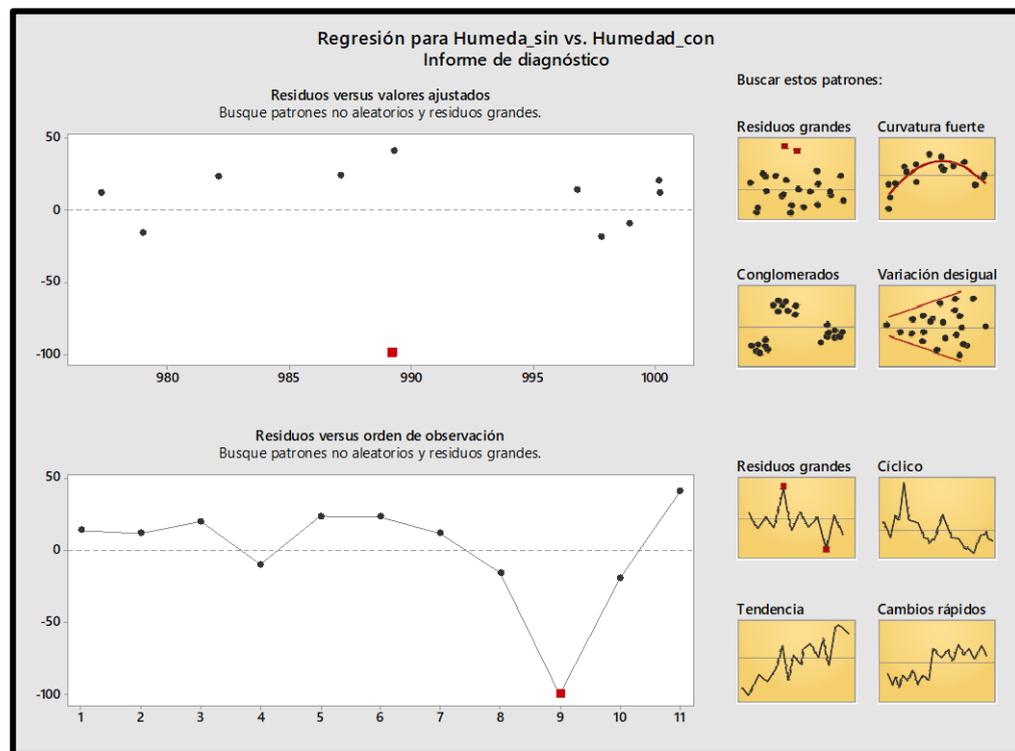
$$t_{(1-\alpha)(n-1)} = 1.81246$$



FUENTE: elaboración propia

Figura 34 — Gráfico de región crítica o aceptación

### f) Interpretación



FUENTE: Minitab v5

Figura 35 — Gráfico de dispersión de porcentaje de humedad con y sin el software



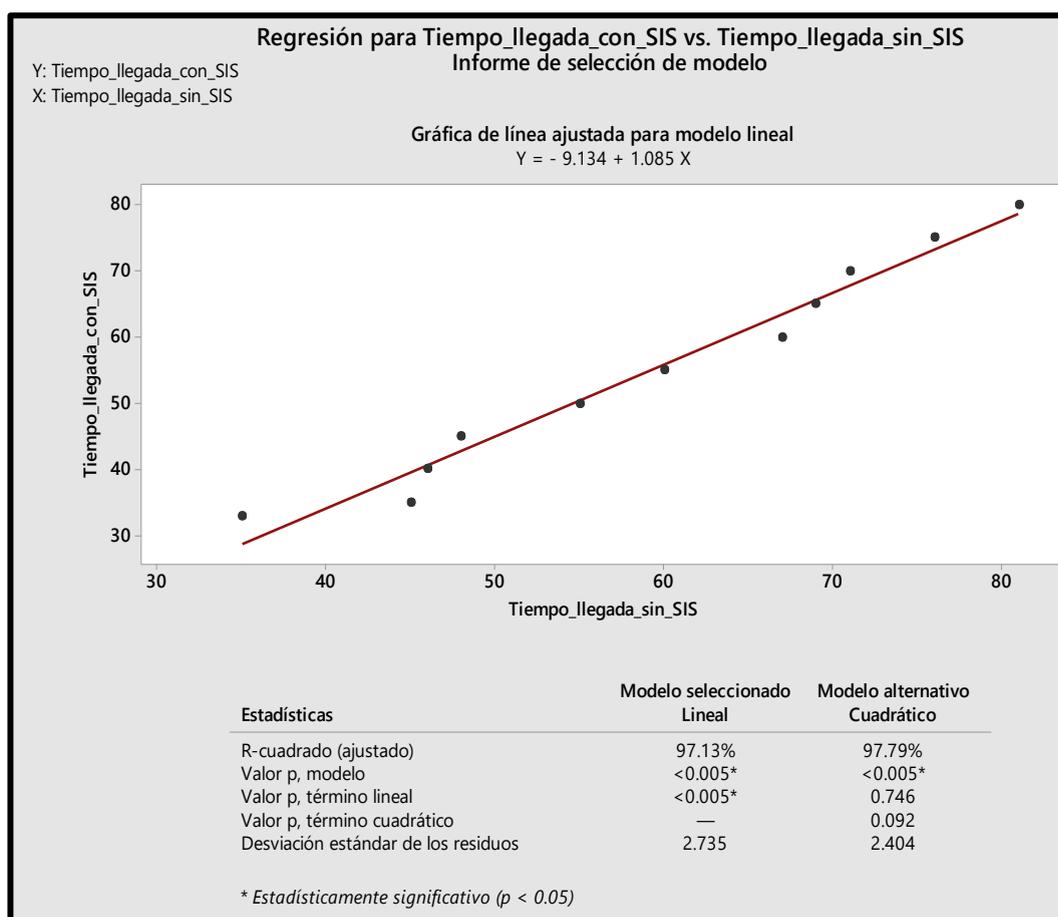
La figura 35 muestra la dispersión de porcentaje de humedad con y sin el software.

Como  $t_c = 3.7586 \nless -\infty; +1.81 >$  entonces se rechaza la hipótesis  $H_0$ , se confirma que al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, mejorar el nivel de humedad del suelo para el cultivo. Esto se debe a que el nivel de humedad es menor sin la aplicación del software de riego tecnificado por goteo.

### 5.3 Discusión

Por medio de los resultados obtenidos con respecto al objetivo general, mejorar la eficiencia de riego para el cultivo mediante la automatización del riego tecnificado por goteo a través del desarrollo e implementación de un software con tecnología Arduino, 2021, se logró una mejora significativa en la eficiencia de riego. Esto se evidencia a través de las pruebas de hipótesis específicas, lo cual confirma que al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, se mejora la eficiencia de riego para el cultivo.

En el procesamiento del objetivo general se obtuvo una correlación del tiempo de llegada del agua a los últimos goteros con la aplicación y sin la aplicación. En la figura 36, se muestra que  $R^2 = 0.9742$  cercano a 1 esto nos indica que el tiempo de llegada aplicando el sistema y sin aplicar el sistema de riego son muy semejantes es un indicador que se tiene el control de los tiempos de riego al momento de realizar el riego por goteo.

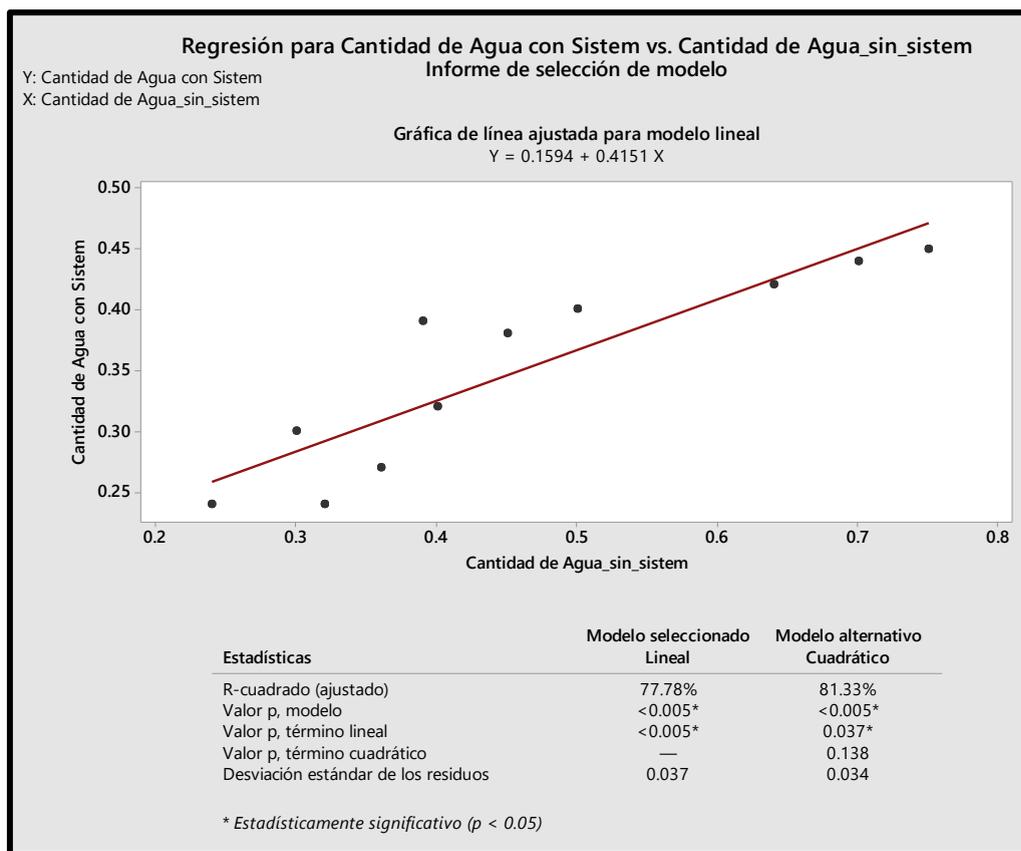


FUENTE: Minitab V5

**Figura 36 — Regresión lineal del tiempo de llegadas del agua a los vegetales**

Respecto al objetivo específico 1, sobre el consumo de agua al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021, el consumo del agua es menor. Por lo que mejora el consumo del agua. En la figura 37, se muestra la gráfica de regresión lineal de la cantidad de agua con y sin la aplicación, donde  $R^2 = 0.80$  un poco lejano a 1 esto nos indica que el control del agua con la aplicación es diferente al control de agua sin la aplicación, por lo que se visualiza una dispersión de los datos del primero con el segundo. Al reducir el consumo del agua, se hace un uso eficiente del agua. Este hallazgo guarda relación con lo que sostiene VILCHEZ CHUPTITAZ (2020), donde “se comprobó que el diseño e implementación del sistema de control automatizado del riego de agua por goteo contribuye significativamente en la disponibilidad del recurso hídrico” (p. 71) .





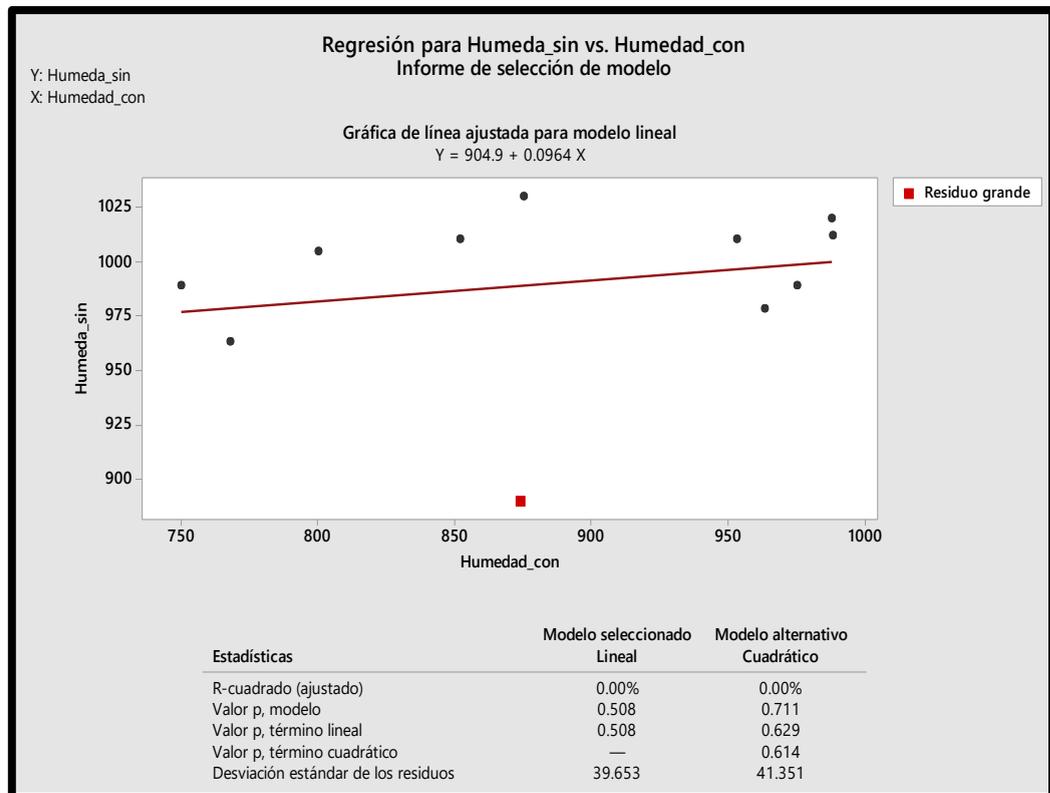
FUENTE: Minitab V5

**Figura 37 — Regresión lineal de la cantidad de agua con y sin la aplicación**

Con respecto al objetivo específico 2, se pudo observar que al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021, se logra una mejora en el nivel de humedad. En la Figura 38, se visualiza en color rojo los datos de control de humedad sin la aplicación del software, mientras que en color azul se presentan los datos de control de humedad con la aplicación del software. Esto indica que el control de humedad es mejor al automatizar el riego tecnificado por goteo mediante el uso del software con tecnología Arduino. Este resultado guarda relación con CASTILLO MELGAR (2021), quien también “diseñó satisfactoriamente un sistema automatizado de riego por goteo para el cultivo de pimiento, utilizando Arduino uno y Tinkercad en función de la humedad del suelo en diferentes texturas a bajos costos y de código abierto o libre” (p. 30). Utilizó sensores de humedad y temperatura, como componente elemental el Arduino uno, que se encargó de controlar todo el funcionamiento del sistema.



Las simulaciones que se realizó demostraron un correcto funcionamiento del sistema automatizado de riego por goteo, para diferentes texturas de suelo, tomando en consideración continuamente la temperatura ambiente, el vegetal en prueba fue diferente a la investigación realizada, sin embargo, los resultados obtenidos fueron similares.



FUENTE: Minitab v5

**Figura 38 — Gráfico del control de humedad del suelo con la aplicación**



## CAPÍTULO VI

### CONCLUSIONES Y RECOMENDACIONES

#### 6.1 Conclusiones

- Al automatizar el riego tecnificado por goteo a través del desarrollo e implementación de un software con tecnología Arduino, 2021, se logró determinar que se mejoró la eficiencia del riego. Esta conclusión se obtuvo mediante la realización de pruebas específicas que fueron llevadas a cabo. Para obtener los resultados deseados, se trabajaron en dos aspectos fundamentales: el diseño agronómico y el diseño hidráulico. El experimento se llevó a cabo utilizando el cultivo de remolachas o betarragas.
- Al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021, se analizó que se logró mejorar el tiempo de riego considerando los valores obtenidos con una desviación estándar con la aplicación de 16.15 y la desviación estándar sin la aplicación de 14.69 que son dos datos muy cercanos. El tiempo de llegada al primer gotero es de 35.0s y al último gotero es de 81.0s. También se observó una distribución desigual en los datos, donde el último gotero requería considerablemente más tiempo para recibir el caudal necesario en comparación con los demás. La diferencia en este aspecto se atribuyó al control más preciso de la cantidad de agua proporcionada por el software, lo que permitió un suministro más eficiente y equitativo.
- Al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021, se determinó que se logró mejorar el consumo de agua. Se logró reducir el consumo del agua en 0.27m<sup>3</sup> por segundo, teniendo en cuenta que la media en cantidad de agua con la aplicación es de 0.3800m<sup>3</sup> y sin la aplicación es de 0.65 m<sup>3</sup> por segundo. Por lo tanto, el consumo de agua es menor con la aplicación en comparación con la ausencia de la aplicación del software de riego tecnificado por goteo, para los cultivos.

- Al automatizar el riego tecnificado por goteo mediante el desarrollo e implementación de un software con tecnología Arduino, 2021, se determinó que se logró mejorar el nivel de humedad en el suelo. El porcentaje de humedad con la aplicación tiene una media de 27.0 y sin la aplicación la media es de 11.6 datos, observando una gran diferencia que se atribuye al control y manejo de la cantidad de agua y esto ayuda en controlar la humedad.

## 6.2 Recomendaciones

- Se recomienda seguir mejorando el software de automatización de riego tecnificado utilizando la tecnología Arduino, integrando más sensores como la medición de la luz solar, ya que podría contribuir a mejorar la precisión de la aplicación al adaptarse a condiciones ambientales cambiantes. Que el sistema sea más flexible y se pueda adaptar a diferentes tipos de cultivo.
- Se recomienda verificar y ajustar la presión o el caudal en el último gotero para garantizar una distribución uniforme, para que estén debidamente calibrados generando un suministro uniforme de agua a todos los goteros.
- Se recomienda realizar un monitoreo continuo del consumo del agua para asegurar que se mantenga los niveles óptimos, y utilizar dichos datos para realizar los ajustes en la programación y configuración del sistema de riego según las necesidades específicas del cultivo.
- Se recomienda llevar a cabo un monitoreo constante del nivel de humedad en el suelo utilizando sensores adecuados. Así como, ajustar los parámetros en el software con tecnología Arduino, para que se encuentre el balance óptimo que garantice un nivel adecuado de humedad en el suelo.



## REFERENCIAS BIBLIOGRÁFICAS

- ABREU, José Luis, 2014. El Método de la Investigación Research Method. *Daena: International Journal of Good Conscience*. Vol. 3, 195-204.
- ANTUNEZ B., A. y FERLMER E., S. 2009. Boletín del Nodo Tecnológico de Riego en el Secano: Eficiencia de Riego en Sistemas Localizados. Instituto de Investigaciones Agropecuarias.
- BALSECA CHISAGUANO, Evelyn Amparo, 2014. Evaluación de Calidad de Productos Software en Empresas de Desarrollo de Software Aplicando la Norma Iso/Iec 25000. (Tesis). Escuela Politécnica Nacional. Quito - Ecuador:
- BENITEZ LLANQUE, Jackeline y FLORES ARN, Alodia, 2013. *Sistema Web Basado en la Norma ISO/IEC 25010:2010 (Square) para la Gestión de Información en el Colegio de Ingenieros del Perú Cd Puno*. Universidad Nacional del Altiplano, PUNO.
- BEHAR RIVERO, Daniel Salomón, 2008. *Metodología de la investigación*. Shalom. ISBN 978-959-212-783-7.
- CAHUEÑAS CALDERÓN, Isaac Vicente, 2017. *Calidad del producto en el desarrollo de software en la Universidad Israel aplicando normas ISO*. Universidad tecnológica Israel. <http://repositorio.uisrael.edu.ec/handle/47000/1474>
- CAICEDO PEDRERA, Antonio, 2017. *Arduino para principiantes*. Turin : IT Campus.
- CASTILLO MELGAR, C. A. (2021). Diseño de un sistema de riego automatizado para cultivos de ciclo corto con Arduino. Estudio de caso pimiento. *Tesis*. Universidad Estatal Península de Santa, La Libertad, Ecuador.
- CAÑON QUIROGA, Diana y CIFUENTES CAMACHO, Edna Luz. (2018). Prototipo de un Sistema Automatizado de riego para Jardines. (Tesis) Fundación Universitaria Los Libertadores. Bogotá Colombia.
- CRUZ LEÓN, José Manuel y BIELSA CALLAU, Jorge, 2001. Sobre la eficiencia, el uso sostenible del recurso del agua y la gestión del territorio Ager. *Revista de Estudios sobre Despoblación y Desarrollo Rural, núm. 1*, pp. 85-112.
- CHANG WONG, L. I. (2020). Diseño de un sistema automatizado de riego por goteo para aumentar la producción de maíz en la Hacienda Durand. *Tesis*. Universidad Católica Santo Toribio de Mogrovejo, Chiclayo.
- DE LOS ÁNGELES MEJÍA, Pedro José. 2000. *Diseño de parque municipal en Santa Cruz de los Cárnavos*. Madrid, ES. 150 p.



- ESCOBAR MANZABA , César Dario y FARFÁN ORELLANA, Karina Jazmín, 2018. *Diseño de un sistema de riego para la implementación de cultivos automatizados en el recinto playa seca del cantón el triunfo*. (Tesis de pregrado). Universidad Agraria del Ecuador. El Triunfo, Ecuador.
- HERNÁNDEZ SAMPIERI, Roberto, FENÁNDEZ COLLADO, Carlos y BAPTISTA LUCIO, Maria Del Pilar, 2014. *Metodología de la Investigación*. 6ta edición. México: Mc Graw Hill. ISBN 978-607-15-0291-9.
- INTAGRI,
- ISO/IEC 25010, 2008. *Software engineering –Software product Quality Requirements and Evaluation (SQuaRE)– Data Quality*.
- ISO/IEC 25000, 2011. *System and software engineering – Systems and software product Quality Requirements and Evaluation (SQuaRE) – Evaluation Process*, International Organization for Standardization. Geneva, Switzerland.
- LATHA, N. Anju, MURTHY, B. Rama y KUMAR, K. Bharat, 2016. Distance sensing with ultrasonic sensor and Arduino., *International Journal of Advance Research, Ideas and Innovations in Technology* Vol. 2
- GUTIÉRREZ QUISPE, Roger David y RAMOS ENCISO, Leonidas. 2021. *Diseño del reservorio de concreto armado para el mejoramiento de riego en Comunidad de Vito Provincia Antabamba Región Apurímac*. Abancay : Universidad Cesar Vallejo, 2021.
- JARA, J. y VALENZUELA, A. 2019. *Formulación de Proyectos de riego localizado para la Ley 18.450*. Universidad de Concepción.
- MANTEROLA, C., Asenjo-Lobos, C. y Otzen, T. 2014. *Hierarchy of evidence:Levels of evidence and grades of recommendation from current*. Chile : Revista Chilena, 2014.
- MARIN CHAMAN, Edson Roberto, 2021. Evaluación de la calidad de producto de software bajo normas ISP/IEC 25000: Caso de estudio sistema de planillas de la Municipalidad Provincial de Chicloayo. Tesis de Pregrado. Perú. Universidad Señor de Sipán. <https://repositorio.uss.edu.pe/handle/20.500.12802/8086>
- MEDINA SAN JUAN, José A, 2000. *Riego por Goteo*. 4th ed. Barcelona, España: Ediciones Mundi Prensa
- MENDOZA, J. (2013). Importancia de la Evaluación de la Eficiencia de los Sistemas de Riego. <https://www.intagri.com/articulos/agua-riego/importancia-de-la-evaluacion-de-la-eficiencia-de-los-sistemas-de-riego>
- NIETO, P. B. y LUENGO, G. R. 2015. *R. Módulo V. Evaluación y síntesis de la evidencia*.



*Guiasalud.es, Biblioteca de Guías de Práctica Clínica del Sistema Nacional de Salud.*

España : s.n., 2015.

OBLITAS CABRERA, Gian Jhair y PÉREZ HERNANDEZ, Luis Eduardo, 2022 “Prototipo de sistema de riego automatizado para mejora de la irrigación en áreas de sembrío de paltas - Olmos” Tesis Pregrado. Perú. Universidad Predo Ruiz Gallo.

PALOMINO, K. 2019. *Riego por aspersión*. Lima : Editorial Macro SAC, 2019.

QUIROGA, Diana Cañón y CAMACHO, Edna Luz Cifuentes, 2018. *Prototipo de un sistema automatizado de riego para jardines*. Bogota d.c : s.n., 2018.

RIVAS, Rene Mario., CANO, Samuel Alejandro y MONTERROZA ALEMAN, Josue Noe. 2022. Sistema de riego automatizado con sensores y Arduino. 1ª ed. San Salvador, El Salv. : Universidad Doctor Andrés Bello.

SANTOS PEREIRA, Luis, 2010. *El Riego Y Sus Tecnologías*. 1ra Edición. España:Editora Europa-Americana ISBN: 13:978-84-692-9979-1

SALINAS ARCOS, Ricardo Miguel, 2019. *Diseño de un prototipo de sistema automatizado con Arduino para riego en el cultivo de fresas.Tesis Doctoral*. Universidad de Guayaquil : Facultad de Ingeniería Industrial. Carrera de Ingeniería en Teleinformática.

SODESSA SOMA, Shonkora y AYODERJI OLALEKAN, Salau (2020). *Modeling and Implementation of an automatic drip irrigation system*. International Conference on Data Analytics for Business and industry.

TAM, Jorge, VERA, Giovanna y OLIVEROS, Ricardo. Tipos. 2008. *Métodos y estrategias de investigación científica. Pensamiento y acción*. España : s.n., 2008.

TORRENTE, Artero Oscar, 2013. *Arduino Caso Practico*. España : RC Libros, 2013.

VÍLCHEZ CHUMPITAS, Hans Aaron, 2020. *Diseño e implementación de un sistema de riego por goteo automatizado para una hectárea de cultivo de durazno en el distrito de La Isla - Asia, Cañete*. Repositorio institucional - URP. <http://repositorio.urp.edu.pe/handle/URP/3354>

YARA. (2024). Producir más, con menos: Eficiencia del uso de agua en riego. <https://www.yara.com.gt/noticias-y-eventos/boletines/producir-mas-con-menos-eficiencia-del-uso-de-agua-en-riego/>



## Anexos

### Anexo 01 — Proceso de identificación y preparación del terreno



FUENTE: elaboración propia

**Figura 39 — Terreno**



FUENTE: elaboración propia

**Figura 40 — Preparación de terreno**



FUENTE: elaboración propia

**Figura 41 — Sembrado de remolacha**





FUENTE: elaboración propia

**Figura 42 — Diseño de las laterales y las subunidades**



FUENTE: elaboración propia

**Figura 43 — Materiales para la instalación del riego**



FUENTE: elaboración propia

**Figura 44** — Goteros empalmes y tubería HDP de ½



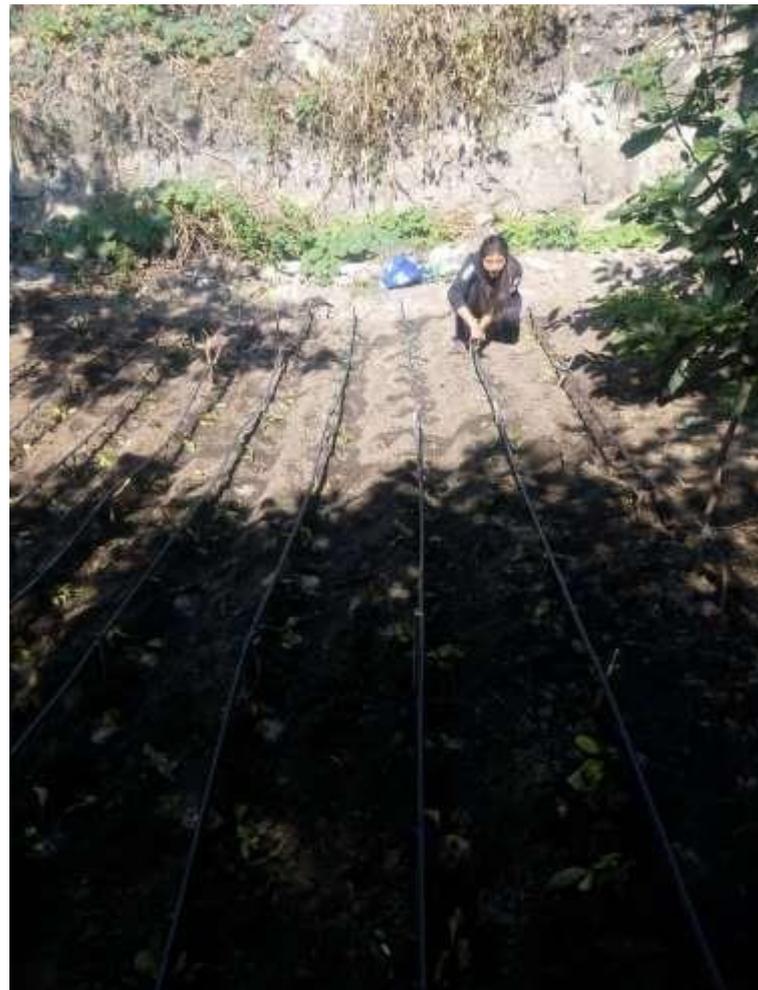
FUENTE: elaboración propia

**Figura 45 — Surcos elaborados de acuerdo al diseño agronómico e hidráulico.**



FUENTE: elaboración propia

**Figura 46 — Distribución de las subunidades del sembrío**



FUENTE: elaboración propia

**Figura 47 — Pruebas del riego por goteo con las subunidades**



FUENTE: elaboración propia

**Figura 48 — Distribución de los goteros de acuerdo al manual de la FAO**



FUENTE: elaboración propia

**Figura 49 — Distribución de los goteros en remolachas**