

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



TESIS

Sistema informático de control y monitoreo para ahorro de agua en cultivos de pastizales en el fundo la peña de la universidad nacional pedro ruiz gallo

Presentado para optar el título Profesional de
Ingeniero en Computación e Informática

Presentado por:

Bach. López Acosta Elmer Arturo

Bach. Vallejos Aguirre Arnold Hidalgo

Asesor:

Dr. Ing. Gilberto Carrión Barco


Lambayeque - Perú 2023

Sistema informático de control y monitoreo para ahorro de agua en cultivos de
pastizales en el fundo la peña de la universidad nacional pedro ruiz gallo

Tesis para optar el título Profesional de Ingeniero en computación e
Informática, que presentan:



López Acosta Elmer Arturo
Tesista



Vallejos Aguirre Arnold Hidalgo
Tesista


ASESORADO POR:



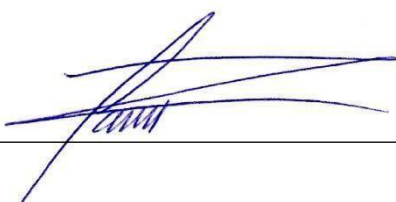
Dr. Gilberto Carrión Barco

Sistema informático de control y monitoreo para ahorro de agua en cultivos de
pastizales en el fundo la peña de la universidad nacional pedro ruiz gallo

APROBADO POR:



Dr. Ing. Segundo Pedro Fiestas Rodriguez
Presidente



M.sc. Janet del Rosario Aquino Lalupù
secretario



M.sc. Ing. Carlos Alberto Valdivia Salazar
Vocal

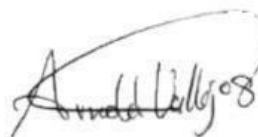
DECLARACIÓN JURADA DE ORIGINALIDAD

Nosotros, López Acosta Elmer Arturo y Vallejos Aguirre Arnold Hidalgo investigadores principales, y el Dr. Gilberto Carrión Barco, asesor del trabajo de investigación “Sistema Informático de Control y Monitoreo para ahorro de agua en Cultivos de Pastizales en el Fundo la Peña de la Universidad Nacional Pedro Ruiz Gallo”, declaramos bajo juramento que este trabajo no ha sido plagiado, ni contiene datos falsos. En caso se demostrará lo contrario, asumo responsablemente la anulación de este informe y por ende el proceso administrativo a que hubiera lugar. Que pueda conducir a la anulación del título o grado emitido como consecuencias de este informe.

Lambayeque, 22 de Septiembre 2023



López Acosta Elmer Arturo
Tesista



Vallejos Aguirre Arnold Hidalgo
Tesista



Dr. Gilberto Carrión Barco
Asesor

DEDICATORIA

*A mis padres, Rosario Acosta y
Alberto López y, a toda mi familia.*

*Porque con sus consejos y
oraciones y palabras de aliento
hicieron de mí una gran persona y
de una u otra forma me acompañan
en todos mis sueños y metas.*

Elmer Arturo López Acosta

A mis padres y a mis hermanas

*Quienes con su amor, paciencia y
esfuerzo me han permitido llegar a
cumplir hoy un sueño más, gracias por
inculcar en mí el ejemplo de esfuerzo y
perseverancia.*

Arnold Hidalgo Vallejos Aguirre

AGRADECIMIENTO

*A Dios, por ser mi guía y acompañante
en el transcurso de mi vida,
brindándome paciencia y sabiduría
para culminar mis metas y mis logros
propuestos*

*A mis padres, a quienes hoy puedo decir:
promesa cumplida; gracias por darme la
mejor herencia, mi profesión.*

*A nuestro asesor, Ing. Gilberto Carrión
Barco, por guiarnos desde el inicio en
este trabajo de investigación.*

Elmer Arturo López Acosta

*A mis padres, por ser mi pilar
fundamental y haberme apoyado
incondicionalmente, pese a las
adversidades de la vida.*

*A nuestro asesor, Ing. Gilberto Carrión
Barco, por guiarnos desde el inicio en
este trabajo de investigación.*

*A los docentes que tuve a lo largo de la
carrera.*

Arnold Hidalgo Vallejos Aguirre

ÍNDICE

RESUMEN	12
ABSTRACT.....	13
INTRODUCCIÓN.....	14
CAPITULO I. DISEÑO TEÓRICO.....	15
1.1. Problemática de la investigación	15
1.1.1. Planteamiento del problema	15
1.2.- Formulación del problema de investigación.....	17
1.3. Hipótesis/solución del problema.....	17
1.4. Objetivos Generales y específicos	17
1.4.1. Objetivos Generales.....	17
1.4.2. Objetivos Específicos	17
1.5. Marco teórico	18
1.5.1. Antecedentes	18
1.6. Bases teóricas	23
1.6.1. Conceptos Generales	23
1.6.1.1. Sistema en Tiempo Real	23
1.6.1.4. Sistema Informático.....	24
1.6.1.5.- Recursos de un Sistema Informático	25
1.6.1.6. Características y Usos de un Sistema Informático.....	26
1.6.1.7. Actividades de un Sistema Informático	27
1.6.1.8. Tipos de Sistemas Informáticos.....	28
1.6.2. Metodología.....	30
1.6.2.1. Metodología Ágil	30
1.6.2.2.- Metodología ágiles Scrum.....	31
1.7.- Plataformas y Arquitecturas	37
1.7.1. Las tecnologías más usadas en la investigación	38
1.7.1.1-Arduino	38
1.7.1.2. Microcontroladores	42
1.7.1.3. Sensores.....	42
1.7.1.4.- Tipos de Sensores	44
1.7.1.5. Actuadores.....	49
1.7.1.6. Periféricos.....	51

1.7.2. Lenguajes de programación.....	54
1.7.2.1.-Lenguaje Arduino	54
1.7.2.2.-Lenguaje C++	55
1.7.2.3.-Java.....	55
1.7.2.4.-Php	56
1.7.3.1.1.- IDE de Arduino.....	56
1.7.3.1.2.- Android Studio.....	57
1.7.4 Sistema de Gestión de Base de Datos MySQL.....	58
1.7.5.-El Riego Tecnificado por Goteo.....	59
CAPÍTULO II: MÉTODOS Y MATERIALES.....	69
2.1.-Diseño de Contrastación de Hipótesis	69
2.2.-Poblacion y Muestra.....	71
2.2.-Técnicas e instrumentos de la recolección de datos.....	72
CAPÍTULO III: RESULTADOS Y DISCUSIÓN.....	73
3.1.-Resultados	73
3.1.2.-Transferencia Bluetooth	79
3.1. 4.1-Implementacion del hosting	103
3.2. Discusión	109
CAPÍTULO IV: CONCLUSIONES.....	111
CAPÍTULO V: RECOMENDACIONES.....	112
REFERENCIAS	113
ANEXOS	120

ÍNDICE DE TABLAS

Tabla 1 Roles y Productos en Crystal	36
Tabla 2 Partes de la placa Arduino UNO	40
Tabla 3 Ventajas y desventajas del Riego por goteo	60
Tabla 4 Tipos de cultivos	66
Tabla 5 Variable Dependiente	70
Tabla 6 Variable Independiente	70
Tabla 7 Población	71
Tabla 8 Muestra	72

ÍNDICE DE FIGURAS

Figura 1 Elementos de un sistema de información	25
Figura 2 Actividades de un sistema informático	27
Figura 3 Proceso Ágil	30
Figura 4 Equipo de desarrollo de la Metodología Scrum	32
Figura 5 Marco de trabajo de la Metodología XP	33
Figura 6 Citricidad de la metodología Crystal	35
Figura 7 Placa Arduino UNO	40
Figura 8 DTH22, sensor de temperatura y humedad	45
Figura 9 Higrómetro, módulo SKU	46
Figura 10 LDR (sensor de luminosidad)	47
Figura 11 Sensor de lluvia	48
Figura 12 Modulo Bluetooth HC – 06	49
Figura 13 Bomba de líquido	50
Figura 14 Electroválvula	50
Figura 15 Relay	51
Figura 16 Display	52
Figura 17 Diodo LED	53
Figura 18 Microprocesador	54
Figura 19 IDE de Arduino	57
Figura 20 Android Studio	58
Figura 21 Riego por goteo	60
Figura 22 Suelo arenoso	61
Figura 23 Suelos calizos	62
Figura 24 Suelos Limosos	63
Figura 25 Suelos Humíferos	63
Figura 26 Suelos Arcillosos	64
Figura 27 Suelos Pedregosos	65
Figura 28 Suelo de turba	66
Figura 29 Arquitectura en Proteus	74
Figura 30 Librería y variables	75
Figura 31 Comunicación serial	76
Figura 32 Funcionamiento del sensor de lluvia	77
Figura 33 Sensor de humedad	77

Figura 34 <i>Suelo húmedo</i>	78
Figura 35 <i>Suelo muy Húmedo</i>	78
Figura 36 <i>Sensor dth</i>	79
Figura 37 <i>Transferencia bluetooth</i>	80
Figura 38 <i>Conexión bluetooth</i>	81
Figura 39 <i>List View</i>	81
Figura 40 <i>Verificación de la BD</i>	82
Figura 41 <i>Llenado de listas</i>	83
Figura 42 <i>Dirección Mac</i>	84
Figura 43 <i>Permiso al usuario</i>	84
Figura 44 <i>Dispositivos a vincular</i>	85
Figura 45 <i>Dispositivo SESORRBT</i>	86
Figura 46 <i>Pantalla inicio</i>	87
Figura 47 <i>Interfaz registro</i> Figura 48 <i>Registro usuario</i>	88
Figura 49 <i>Registro de datos</i>	89
Figura 50 <i>Logeo</i>	90
Figura 51 <i>Menú opciones</i>	91
Figura 52 <i>Botón registro/ bienvenida</i>	92
Figura 53 <i>Librerías Botón Control</i>	93
Figura 54 <i>Declarando variables y UUID</i>	93
Figura 55 <i>Método Handle</i>	94
Figura 56 <i>Conexión bluetooth usando UUID</i>	94
Figura 57 <i>Obtener listado de dispositivos Bluetooth</i>	95
Figura 58 <i>Creación de eventos</i>	96
Figura 59 <i>Creación de hilos</i>	96
Figura 60 <i>Mensaje Conexión</i>	97
Figura 61 <i>Código XML del botón control</i>	97
Figura 62 <i>Control Arduino</i>	98
Figura 63 <i>Resultado suelo seco</i>	99
Figura 64 <i>Resultado suelo húmedo</i>	100
Figura 65 <i>Resultado suelo muy húmedo</i>	101
Figura 66 <i>Resultado sistema apago</i>	102
Figura 67 <i>Hosting libre 000webhost</i>	103
Figura 68 <i>Administrador de Archivos</i>	104
Figura 69 <i>Subir Archivos BD</i>	105
Figura 70 <i>Conexión. Php</i>	106
Figura 71 <i>Logeo.php</i>	107
Figura 72 <i>Register.php</i>	107
Figura 73 <i>Base de datos</i>	108
Figura 74 <i>PhpMyAdmin</i>	108
Figura 75 <i>Selección de opción SOFTWARE</i>	122
Figura 76 <i>Selección de descarga según S.O</i>	122
Figura 77 <i>Descarga de Support the Arduino IDE</i>	123
Figura 78 <i>Inicio de descarga</i>	123
Figura 79 <i>Ejecutable Arduino IDE</i>	123
Figura 80 <i>Acuerdo de licencia</i>	124
Figura 81 <i>Instalación Arduino IDE</i>	124

Figura 82 <i>Instalacion paso 2 Arduino IDE.....</i>	125
Figura 83 <i>Instalación paso 3 Arduino IDE.....</i>	125
Figura 84 <i>Seleccionamos icono.....</i>	126
Figura 85 <i>Búsqueda de librería.....</i>	126
Figura 86 <i>Instalación de librería.....</i>	127
Figura 87 <i>Librería instalada.....</i>	127
Figura 88 <i>Interfaz principal de Proteus.....</i>	128
Figura 89 <i>Descarga paso 1.....</i>	128
Figura 90 <i>Descarga paso 2.....</i>	129
Figura 91 <i>Verificación de email.....</i>	129
Figura 92 <i>Enlace de descarga.....</i>	130
Figura 93 <i>Inicio de descarga.....</i>	130
Figura 94 <i>Ejecutable Proteus 8 Professional.....</i>	130
Figura 95 <i>Primer requisito paso 1.....</i>	131
Figura 96 <i>Primer requisito paso 2.....</i>	131
Figura 97 <i>Segundo requisito paso 1.....</i>	132
Figura 98 <i>Segundo requisito paso 2.....</i>	132
Figura 99 <i>Instalación paso 1 Proteus 8.....</i>	133
Figura 100 <i>Instalación paso 2 Proteus 8.....</i>	133
Figura 101 <i>Instalación paso 3 Proteus 8.....</i>	134
Figura 102 <i>Instalación paso 4 Proteus 8.....</i>	134
Figura 103 <i>Página principal de Android Studio.....</i>	135
Figura 104 <i>Inicio de descarga Android Studio.....</i>	135
Figura 105 <i>Ejecutable Android Studio.....</i>	136
Figura 106 <i>Paso 1 Instalación Android Studio.....</i>	136
Figura 107 <i>Paso 2 Instalación de Android Studio.....</i>	137
Figura 108 <i>Paso 3 Instalación de Android Studio.....</i>	137
Figura 109 <i>Paso 4 Instalación de Android Studio.....</i>	138
Figura 110 <i>Paso 5 Instalación de Android Studio.....</i>	138

RESUMEN

En la actualidad el fundo La Peña de la Universidad Nacional Pedro Ruiz Gallo, hace el mal uso de los recursos hídricos, por la cual emplean una gran cantidad de agua en el riego, ocasionando la pérdida de los cultivos de los pastizales, de manera que se desarrolló un sistema informático de control y monitoreo, para mejorar el riego y ahorro del agua en el cultivo, la investigación de este proyecto es de tipo pre-experimental aplicada a los conocimientos básicos de la electrónica y la tecnología IoT, para la recolección de datos se utilizó un cuestionario y una ficha de registro de datos. Para realizar este proyecto se optó por utilizar tres tipos de sensores, las cuales son: sensor de humedad de suelo, sensor de lluvia y sensor de temperatura, al tener un sensor de Humedad de suelo, cumplirá la función de monitorear, este transmitirá mediante una señal analógica, a la placa Arduino y este a su vez procesará la información, para poder generar una señal de salida, de esta manera mandará una señal al relay, que activará o desactivará la bomba de agua, como resultado obtenido tuvimos una mejora en la distribución del agua en los cultivos, aplicando el riego tecnificado por goteo. En la que se desarrolló una aplicación móvil en Android Studio, el cual nos permite el monitoreo en tiempo real, y la recepción de información, utilizando el sistema de gestión de datos MYSQL.

Palabras claves: Riego y ahorro de agua, humedad del suelo, Sistema informático de control y monitoreo.

ABSTRACT

At present, the La Peña farm of the Pedro Ruiz Gallo National University misuses water resources, for which they use a large amount of water in irrigation, causing the loss of pasture crops, so that A control and monitoring computer system was developed to improve irrigation and save water in the crop, the research of this project is of a pre-experimental type applied to the basic knowledge of electronics and IoT technology, for the collection of data, a questionnaire and a data record sheet were used. To carry out this project, it was decided to use three types of sensors, which are: soil humidity sensor, rain sensor and temperature sensor, having a soil humidity sensor, it will fulfill the function of monitoring, it will transmit through a analog signal, to the arduino board and this in turn will process the information, in order to generate an output signal, in this way it will send a signal to the relay, which will activate or deactivate the water pump, as a result obtained we had an improvement in the distribution of water in crops, applying drip irrigation technology. In which a mobile application was developed in Android Studio, which allows us to monitor in real time, and receive information, using the MYSQL data management system.

Keywords: Irrigation and water saving, computer control and monitoring system.

INTRODUCCIÓN

El agua en la actualidad es un recurso imprescindible, que es limitado tanto en cantidad como calidad por el incremento acelerado de la demanda para uso doméstico e industrial, por lo que en los sistemas de riego debe utilizarse de manera más eficiente.

La agricultura es aquel régimen con una elevada demanda de agua a nivel mundial, donde la actividad con un 70% de consumo de agua es el riego, tan solo se calcula que el 20% de la tierra agrícola global se encuentran bajo riego, siendo aquellas las que suministran un 40% en de manera general de la producción agrícola.

Con la finalidad de ayudar en el riego tradicional que se emplean en algunos sectores de la región Lambayeque, se está optando por implantar la tecnología para planificación, el manejo y programación del riego considerando el periodo de crecimiento de los pastizales, como consecuencia se decidió hacer un sistema informático, el cual lleva por título “Sistema informático de control y monitoreo para ahorro de agua en cultivos de pastizales en el fundo La Peña de la Universidad Nacional Pedro Ruiz Gallo”

Por el cual, se implementó un prototipo basado en Arduino, con ayuda de diferentes tipos de sensores de humedad de suelo, lluvia y temperatura, lo cual cumple la función de automatizar el riego y poder hacer un buen control en el ahorro de agua. También se desarrolló un aplicativo móvil el cual permitirá que el usuario pueda visualizar y monitorear las funciones que realiza nuestro sistema informático.

CAPITULO I. DISEÑO TEÓRICO

1.1. Problemática de la investigación

1.1.1. Planteamiento del problema

La situación problemática a nivel mundial, por medio recursos de agua dulce disponibles per cápita han decaído en más del 20% en los postremos 20 años por el crecimiento de la población y al impulso exacerbado de la economía por el cambio climático y, a menos que se tomen medidas, la tendencia continuará, refieren las autoridades agrarias en reunión de las Naciones Unidas. Según el principal análisis del estado global de la alimentación y la agricultura de la Agencia Mundial, más de 3 000 000 000 de individuos residen en tierras agrícolas con estrés hídrico severo, de las cuales casi la mitad, 1 200 millones, están sujetas a restricciones (ONU, 2020).

El uso de agua per cápita cayó un 22 % en América Latina, un 27 % en Asia sur y hasta un 41 % en el África subsahariana, donde esta última cerca de 50 000 000 de individuos habitan en áreas donde las sequías severas cada tres años pueden tener efectos devastadores en las tierras de cultivo y los pastizales. Alrededor del 11% o 128 millones de hectáreas de las tierras agrícolas de secano del mundo dependen únicamente del agua de lluvia y no tienen sistema de riego, por lo que se ven afectadas con frecuencia por la sequía. Mientras tanto, más del 60% de las tierras agrícolas de regadío, o más de 171 000 000 de hectáreas, sufren estrés hídrico severo. según el informe, emplear una administración adecuada del agua, una atribución clara, tecnología moderna y un cambio hacia cultivos con menores requerimientos de agua es urgente y necesario. En general, el 20% de las tierras agrícolas, unos 300 000 000 de hectáreas, sufre escasez de agua. También cabe señalar que el acceso per cápita al agua es desigual en todo el mundo. Por ejemplo, en 2017 Oceanía promedió alrededor de 43 000

metros cúbicos de agua dulce por persona, mientras que el Norte de África y Asia Occidental alcanzaron poco menos de 1000 metros cúbicos.

En nuestro país la agricultura es la principal actividad que asisten al desarrollo del país, según la CCL “esta actividad representa aproximadamente el 5,4% del PBI nacional” y se brindó oportunidades de trabajo a más de 4 millones de compatriotas en el año 2019 (Antón, 2020). El problema principal en nuestro país de insostenibilidad tiene que ver con la diferencia de distribución de territorio del agua. El Perú ofreció mecanismos de inversión en proyectos de irrigación hidráulica que favoreció la Región Costa, disminuyendo el apoyo a las otras regiones del país. ENAHO la Encuesta Nacional de Hogares, refiere que solamente el 41% de las unidades agropecuarias familiares de subsistencia cuentan con riego a diferencia del 93% de la agricultura empresarial.

El fundo La Peña de la UNPRG, desempeña el rubro de sembrío y venta de cultivos teniendo más de 12 años de funcionamiento, tiene convenio con el ministerio de agricultura, a su vez posee un área de 05 hectáreas de sembrío, donde se encuentran cultivos de alfalfa, moringa, pasto elefante, pasto alemán, maíz, limas, naranjas, espárragos, algodón.

Dichos cultivos podrían llegar a ser afectados por la gran cantidad de agua que se realiza en el riego de las parcelas, provocando el mal uso del recurso hídrico, ya que si se llega a exceder, la temperatura del suelo puede llegar a un cierto punto de humedad pudiendo ocasionar la descomposición del cultivo, originando la pérdida de la producción final, es por ello que para poder evitar dichos problemas mencionados se diseñara un Sistema de riego por goteo, que beneficiara el ahorro del agua en el riego de los cultivos dándonos unos buenos resultados en el riego de los suelos.

1.2.- Formulación del problema de investigación

¿De qué manera el sistema Informático de control y monitoreo permitirá mejorar el riego y ahorro del agua en los cultivos de pastizales en el fundo la Peña de la Universidad Nacional Pedro Ruiz Gallo?

1.3. Hipótesis/solución del problema

Mediante un sistema informático de control y monitoreo se mejorará el riego y ahorro de agua, en el cultivo de pastizales del fundo la Peña de la Universidad Nacional Pedro Ruiz Gallo.

1.4. Objetivos Generales y específicos

1.4.1. Objetivos Generales

Desarrollar un sistema informático de control y monitoreo, para mejorar el riego y ahorro del agua en el cultivo de pastizales en el fundo la Peña de la Universidad Nacional Pedro Ruiz Gallo.

1.4.2. Objetivos Específicos

- Diagnosticar el estado actual de la forma de regadío en el fundo de la Universidad Nacional Pedro Ruiz Gallo.
- Diseñar el circuito electrónico utilizando Proteus.
- Elaborar un prototipo para el control y monitoreo del sistema de riego utilizando la tecnología Arduino y el uso de sensores.
- Crear una pequeña base de datos en MySQL, para permitir el logeo de los usuarios en la aplicación móvil.

- Elaborar una aplicación móvil, desarrollado en Android, que permita la recepción de los valores de los sensores y su visualización.

1.5. Marco teórico

1.5.1. Antecedentes

Antecedentes Internacionales

Según Barroso (2015), obtuvo como resultado la toma de datos, permitiéndole simular las condiciones ambientales del lugar en el interior de un invernadero. En la programación del Arduino se realice siguiendo una programación basada en la orientación a objetos, con lo cual les facilita realizar nuevas medicaciones. En el desarrollo de la aplicación Android ofrece un control de interfaces fácil de manejar por el usuario, siendo compatible con las diferentes versiones del S.O. Se ha realizado todos los procesos del desarrollo propuesto consiguiendo el objetivo principal del proyecto que era “implementar una aplicación remota, capaz de efectuar un control y una monitorización sobre un invernadero” a través de una aplicación móvil, otorgando una gran ventaja desde el punto vista económico y con el fácil manejo de la aplicación.

Por su parte Perea (2016), desarrolló la aplicación que permita el monitoreo, registro y control de un cultivo en invernadero mediante la implementación de diferentes dispositivos electrónicos, desarrollado el sistema en un bajo costo y de fácil manejo lo cual le permitió brindarle al usuario un mayor control en las mediciones de las variables físicas más relevantes como la temperatura, luminosidad, humedad de suelo y humedad relativa. Su buen funcionamiento de la aplicación tiene una buena comunicación entre su sistema microcontrolado que forman parte de la familia Arduino, el cual tendrá la función de recibir los datos y enviarlos al servidor que le permite poder ver el registro de los valores

encontrados almacenados, con lo cual se creó una aplicación desarrollada en PHP con la finalidad de poder almacenar la información en la base de datos creada en MySQL.

Nos menciona Bustos (2017), que este proyecto se basa en una propuesta de un sistema de riego automatizado para el uso de manera remota con el fin de facilitar el riego de cultivos de lechuga en la finca Los Almendros del departamento de Jinotega. La idea es tener un buen resultado integralmente en el sistema inteligente controlado por una aplicación de Android por medio de un teléfono móvil y utilizando un sensor de humedad de suelo que determinara la cantidad de agua a recolectar asociada a ella y la cantidad de plantas necesarias para optimo crecimiento y desarrollo, por lo que pretende lograr un adecuado manejo de los cultivos y facilitar su correcto desarrollo.

Para la automatización del sistema de riego se hará el uso del micro controlador ATmega328 de la placa Arduino, el cual es la parte esencial para lograr los objetivos requeridos para este proyecto en el fácil uso y aplicación, esta tecnología contribuye directamente con el medio ambiente, dándoles un mejor manejo en sus recursos hídricos. Obteniendo productos de alta calidad con un alza en nuestros tiempos de un mejor control y manejo para conseguir los mejores resultados. La tecnología Arduino planea completar la tarea que deseamos concretar, junto con un diagrama específico será responsable de hacer órdenes para la funcionalidad del sistema de riego. Los usuarios podrán manipular y controlar en cualquier época del año dicho sistema de cultivo sin problema alguno.

Según Escobar (2018), su estudio se basa en la optimización de sistemas de riego para arboles de cacao en una playa seca en el estado de El Triunfo. Dado que los sistemas de riego actuales provocan un consumo excesivo de agua, para mejorar el consumo de agua se ha planteado diseñar un sistema de riego tecnificado por goteo para disminuir la cantidad de recursos hídricos que van dirigidos al riego del cacao. La placa Arduino que utiliza relés para controlar electroválvulas que tienen como finalidad permitir el acceso del agua a través de tuberías para que sean dirigidas a cada zona de riego del cultivo, y donde los sensores como la humedad de suelo y la lluvia miden las variables y hacen la toma de decisiones para el riego solo cuando sea necesario para la planta en modo automático.

Pereo (2018), tuvo como objetivo satisfacer las necesidades existentes de modernización de los sistemas de riego existentes. Estos sistemas no cumplen con las necesidades de programación de los cultivos de próxima generación, como el riego hidropónico, aparte del control de audio deficiente y la incapacidad de interactuar de forma remota con ellos a bajo costo. Dicho prototipo está desarrollado con la tecnología Arduino, cuenta con sensores de temperatura y humedad, sensores de humedad del suelo (un sensor para cada área monitoreada) y la capacidad de un servidor web para conectarse a través de un cable de red Ethernet, lo que facilita su operación e interacción remota con el dispositivo para ver las alarmas registradas por el sensor. El dispositivo permite configurar los valores de alarma de los sensores de temperatura, humedad del aire y humedad del suelo de forma totalmente independiente, así como el horario (cuatros de inicio diferentes), la frecuencia para el riego hidropónico se requiere más de cuatro riegos al día y la duración del riego. En resumen, el auge de los dispositivos IoT se debe en gran medida a que Arduino se encarga de diseñar dispositivos de bajo costo que satisfagan completamente las necesidades existentes en cualquier región, y no solo parcialmente como es el caso de los dispositivos comerciales.

Antecedentes Nacionales

Escala (2014), desarrolló su proyecto teniendo como fin enlazar los sensores de humedad temperatura, lluvia, y flujo de agua a Arduino, programar los sensores para poder recibir los datos en el formato deseado, configurar la Raspberry Pi para recoger los datos de los sensores y lograr conectividad a un punto de acceso, crear un sistema que permita subir los datos a la nube, finalmente ofrecer una interfaz Restful que permita monitorizar las diferentes instalaciones.

Por su parte Quispe (2018), implementando la tecnología de información y comunicación, con la finalidad de mejorar el tiempo de trabajo y ahorro de agua para la empresa. La investigación fue desarrollada bajo el diseño no experimental, transaccional – descriptivo. Para su desarrollo se optó por usar la tecnología Arduino, lo cual le permitió poder plantear un correcto diseño de un sistema de riego automatizado, de forma que ayudo a monitorear la cantidad de humedad que se presentaba en el suelo del vivero donde se realizaron las pruebas. Para optar por este diseño se realizó una encuesta a los trabajadores para plantearles dicho proyecto, el cual dieron como aprobatorio en cambiar la forma de riego tradicional por una como la del riego automatizado con la cual se empleó la aspersión para el riego en las cosechas, brindándoles una mejor simulación en la minimización del tiempo de trabajo y ahorro de agua.

Según Gutierrez (2020), su proyecto se basó en instalar un sistema de riego subterráneo, el cual será controlado por un circuito lógico que controla el flujo de agua, y la salida del circuito de control es alimentada a un elemento actuador electro mecánico. Por consiguiente, una válvula de control accionada mecánicamente permanece abierta al nivel deseado. El circuito calibra el tiempo que tarda en activarse y regar durante la noche. Por esto, se creó un temporizador digital para controlar el tiempo de trabajo. También incluye

un sistema de sensores que mide la potencia de la luz y la humedad del suelo para regular el cierre de un interruptor adicional que fuerza el agua a través de un sistema de tuberías con aberturas lo escasamente pequeñas para mantener el riego por goteo. Utilizando el sistema desarrollado, se puede ahorrar el consumo de agua en el riego.

Romeo (2019), desarrolló un proyecto basado en riego automatizado en plantas de palma aceitera se desarrolló en la planta productiva de la empresa OLAMSA.SA (Oleaginosas Amazónicas) con el fin de crear un método para poder controlar de manera efectiva la cantidad de agua que una plantación de palma requiere diariamente de acuerdo con sus necesidades nutricionales. Este proyecto explica y analiza todas las herramientas, recursos técnicos y conocimientos teóricos para crear un sistema de riego automatizado que sea eficiente haciendo el uso de software y hardware libre para el desarrollo del proyecto final. Este estudio también describe en detalle el método de puesta a punto del sistema de riego automatizado implementado en plantaciones de palma aceitera, sus componentes y resultados en plantones. Por consiguiente, esta tesis brinda conclusiones, evaluaciones, anexos y bibliografías en base al desarrollo y propuesta de sistema de riego automatizado, donde se presenta de forma resumida la factibilidad del proyecto.

Antecedentes Regionales

Según Estrada (2020), la razón principal del uso ineficiente del riego es la falta de un sistema de riego integrado y acelerado, tal motivo que aún se sigo usando el riego de manera antigua en el complejo deportivo “San Juan Masías”, por lo cual tiene como finalidad tener una mejor eficiencia en el riego mediante un sistema presurizado por aspersión que sería un buena opción para el complejo deportivo, aplicando la topografía en el lugar tenemos como resultado un aumento en el consumo de agua en 46.43%, para las 1.09 ha proyectadas así también hay un total de 21 aspersores con un caudal de 2.9 m³/hr, 14 aspersores con un

caudal de 0.54 m³/hr y 18 aspersores con un caudal de 0.5 m³/hr; en parte final es 55.85 m³/ha/día y las longitudes de la tubería primaria y el porta lateral N°1 es de 144 mm, el siguiente N°2 es de 88mm, en el primer turno las pérdidas fueron de 21.902 mca y las siguientes fueron de 30.346 mca; la diferencia de eficiencia entre el método de riego inestable aplicado y el sistema de baja presión propuesto es significativa, aun comparando el área actual regada (0.67 ha) con el área proyectada, hay una mejora del 13.61%, lo que permite entender la importancia que es la tecnología en el riego de cultivos y por ende el buen funcionamiento de los equipos y mantenimiento en consecuencia para alargar la vida útil de estos equipos y con el tiempo su funcionamiento sea la mejor ya que la compra de estos equipos para este sistema es costosa.

1.6. Bases teóricas

1.6.1. Conceptos Generales

En esta sección presentamos los aspectos básicos del sistema informático de control y monitoreo para ahorro de agua en los cultivos como fundamentos, métodos, herramientas y entornos para su desarrollo. Estos conceptos serán esenciales para comprender su aplicación específica en los capítulos posteriores.

1.6.1.1. Sistema en Tiempo Real

Según Antonio,M. (2012) Un sistema de tiempo real es un sistema de procesamiento de información que responderá estímulos de entrada generados externamente en un período de tiempo y funcionamiento. Las respuestas correctas dependen no solo de los resultados lógicos sino también del tiempo en que son entregadas. Los sistemas de tiempo real generan alguna acción en respuesta a sucesos externos. Para realizar esta función, ejecutan una

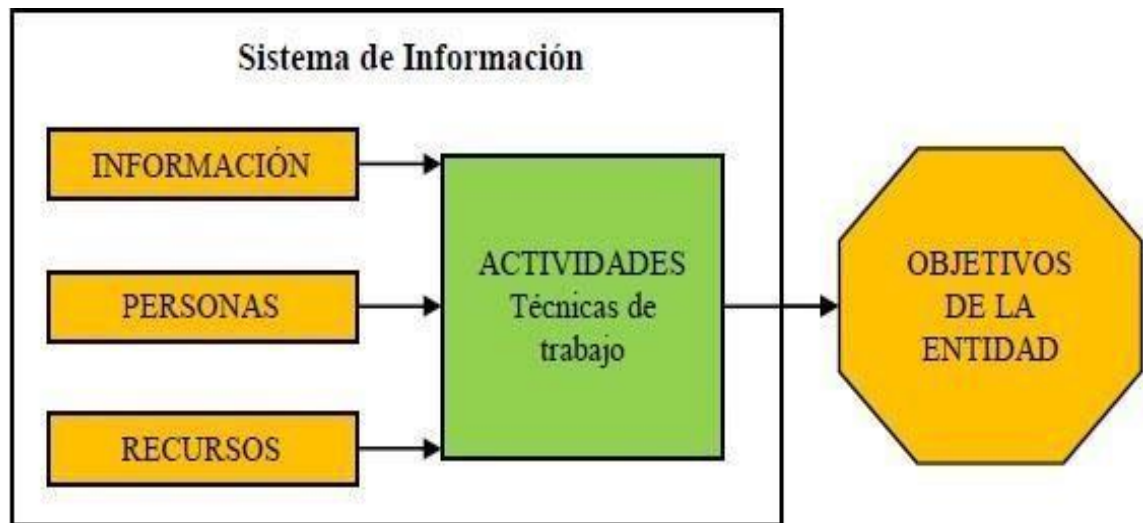
adquisición y control de datos a alta velocidad bajo varias ligaduras de tiempo y fiabilidad. Debido a que estas ligaduras son muy rigurosas, los sistemas de tiempo real están frecuentemente dedicados a una única aplicación. Durante muchos años, los principales consumidores de sistemas de tiempo real eran militares. Sin embargo, hoy las significativas reducciones de los costos del hardware han hecho posible para la mayoría de las compañías proporcionar sistemas (y productos) de tiempo real para diversas aplicaciones, ejemplo que incluyen control de procesos, automatización industrial, investigación médica y científica, gráficos de computadoras, comunicaciones locales y de largo alcance.

1.6.1.4. Sistema Informático

Según Loudon (1999) un sistema Informático está compuesto por un conjunto de elementos que están encaminados al procesamiento y administración de datos e información, estos elementos están organizados y preparados para usarlos después, inventados para cubrir una necesidad. Los elementos que componen a un sistema informático formaran parte de alguna de las siguientes categorías:

- Personas.
- Datos.
- Actividad o Técnicas de trabajo
- Recursos materiales en general (generalmente recursos informáticos y de comunicación, aunque no necesariamente).

Figura 1 Elementos de un sistema de información



Fuente (Loudon, 1999)

Estos elementos dentro de un sistema de información se relacionan con los datos del proceso y surge una información más elaborada y de manera más apropiada se distribuye dentro de una empresa dada de acuerdo a su objetivo.

Según Kendall (2007), al igual que cualquier sistema, un Sistema informático consta de un cúmulo de partes relacionadas, como software hardware y recurso humano (humanware) que permiten a una organización almacenar y procesar información.

1.6.1.5.- Recursos de un Sistema Informático

Un sistema informático está compuesto por los siguientes elementos: software, hardware, recurso humano (FAO, 2005).

- Hardware

La fundación de la comunidad fondo de comercio (GCF, 2005) refiere que el hardware es la parte que se puede observar desde la computadora, es otras palabras son los componentes de su estructura física (monitor, mouse, fuente de poder), son parte del

hardware de la computadora. Por otro lado, la FAO (2015) fortaleció el concepto anterior al referirse al término “hardware” para referirse equipo físico del plan operativo.

- **Software**

Para la fundación de la comunidad fondo de comercio (GCF, 2015) El software son programas informáticos que hacen posible la realización de tareas específicos, como por ejemplo el Word, Excel, PowerPoint, los navegadores, etc. Del mismo modo, la FAO, advierte que la computadora en si no es autónoma, por lo que es necesario un conjunto de instrucciones para decirle a la computadora que operaciones realizar. Por lo tanto, todas estas instrucciones se denominan “programas informáticos” o, en ingles “software”.

- **Recurso Humano**

Según Alegsa (2016), son todas los individuos que son parte del sistema, como ser los operadores del sistema, los técnicos que lo mantienen y los usuarios finales.

1.6.1.6. Características y Usos de un Sistema Informático

Según Stalling (2000) las características de un sistema informático son las siguientes:

- Se logra ahorros considerables en mano de obra.
- Primer tipo de sistema informático que se crea en las empresas.
- Son estrictos en entrada y salida de información; sus procesos son simples, se requiere de mucho manejo para sus operaciones y obtener como resultado grandes base datos de información.

1.6.1.7. Actividades de un Sistema Informático

Según Loudon (1999), el sistema informático realiza tres actividades con la finalidad de generar información en la organización y cooperar con la toma de decisiones, las operaciones de controlar y análisis del problemas. Estas actividades consisten en los siguientes componentes:

- **Entrada**

Captura de datos sin procesar dentro de la organización y su entorno.

- **Procesamiento**

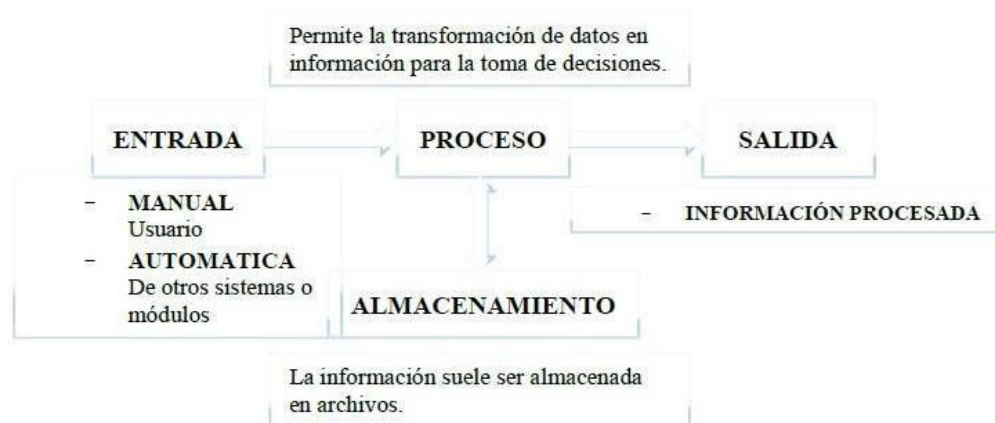
Convierte esa entrada de datos en una forma más significativa.

- **Salida**

Transfiere información procesada a la persona para ser necesaria.

Las tres actividades ejecutadas en un sistema informático interactúan entre sí, tal como se grafica en la figura siguiente.

Figura 2 *Actividades de un sistema informático*



Fuente: Sistema de información (Loudon, 1999).

- **Entrada de datos**

El proceso de captura de datos y preparación para su proceso posterior. Las entradas pueden ser manuales o automáticos.

Almacenamiento de datos

Proceso mediante el cual el sistema almacena de manera organizada los datos e información para su uso posterior.

- **Base de Datos:** Un conjunto de registros interrelacionados, por ejemplo: la base de datos de empleados de la organización, puede incluir archivos de salarios mensuales, así como otros archivos relacionados con los registros de evaluación del desempeño de cada trabajador, la asistencia en la capacitación, etc.

- Procesamiento de Datos

Tiene la capacidad de realizar operaciones con los datos Almacenados en las unidades de memoria.

Salida de Información

Actividad que permite transmitir información útil y valiosa a los usuarios finales.

1.6.1.8. Tipos de Sistemas Informáticos

Según Lázaro (2008)

- **Sistemas de procesamiento básico de la información:**

Estas computadoras están equipadas con el título de trabajo de la información. Las personas que desarrollaron el sistema cubren todos los grifos asociados con la preparación de materiales reales y el análisis de los resultados.

- **Sistema de apoyo a la toma de decisiones:**

Confían en MIS para crear y actualizar bases de datos, que es lo que se usó primero. Los DSS están diseñados para la toma de decisiones, se utilizan para respaldar el trabajo individual o la toma de decisiones en equipo, respaldan una gran cantidad de la llamada investigación de operaciones o métodos cuantitativos en la toma de

decisiones, un tipo muy importante de DSS es la planificación de recursos empresariales. (Sistema ERP)

- **Sistema basado en la inteligencia Artificial:**

La inteligencia artificial, una rama de la informática, busca imitar las capacidades intelectuales del ser humano, utilizando software especializado y computadoras.

Sistemas de conocimiento profesional o sistemas de conocimiento basados en reglas.

- **Plataforma Libre**

Hardware libre: Se llama hardware libre, electrónica libre o maquinas libres a aquellos dispositivos de hardware cuyas especificaciones y diagramas esquemáticos son de acceso público, ya sea bajo algún tipo de pago o de forma gratuita. La filosofía del software libre es aplicable a la de hardware libre y por eso forma parte de la cultura libre. Un ejemplo de hardware libre es la arquitectura UltraSparc cuyas especificaciones están disponibles bajo una licencia libre.

Software libre: (También llamado Free Software) Es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el producto y, por tanto, una vez obtenido el mismo, puede ser usado, copiado, estudiado, modificado y redistribuido libremente de varias formas. Según Free Software Foundation, el software libre se refiere a la seguridad de los usuarios para ejecutar, copiar, distribuir y estudiar el software, e incluso modificarlo y distribuirlo.

1.6.2. Metodología

1.6.2.1. Metodología Ágil

Para la creación del prototipo y aplicación móvil en este proyecto haremos uso de una metodología de desarrollo de software lo cual nos permitirá hacer el desarrollo de manera ágil, ligero, enfocado en las buenas prácticas para cumplir con los objetivos, calidad, productividad y por ende se realizará un “seguimiento diario de los progresos del proyecto”. Scrum forma parte de esta metodología que nos proporcionará una minimización de los riesgos en la ejecución de este trabajo, de forma colaborativa.

Según López (2015) nos dice que el término “ágil” surgió de la idea de un grupo de personas expertas en el área de desarrollo de software con la finalidad de poder optimar el proceso de “creación” del mismo, y este era de manera rígida y con mucha documentación.

Manifiesto Ágil

En el desarrollo de software ágil en un proyecto, este engloba los principios y valores que hacen que sea diferente a uno que sea desarrollado de forma tradicional.

Figura 3 *Proceso Ágil*



Fuente: (Lopez, 2015)

1.6.2.2.- Metodología ágiles Scrum

- **Scrum**

Scrum se basa al control del proceso empírico o a la teoría empírica. El conocimiento proviene de la experiencia y las decisiones basadas en el conocimiento conocido es lo que nos asegura el empirismo. Este método hace el uso de metodologías de manera iterativa e incremental para optimizar la predictibilidad y el control de riesgos, que se ejecute en la entrega del proyecto en sí.

Para el control del proceso empírico existen tres pilares fundamentales los cuales son:

- Transparencia
- Inspección
- Adaptación

Se describen cuatro eventos importantes en la metodología SCRUM:

- Reunión de planificación del sprint (Sprint Planning Meeting).
- Revisión del Sprint (Sprint Review).

La metodología SCRUM tiene como objetivo centrarse en la segmentación del trabajo completo (Product Backlog) en diferentes dispositivos que puede ser considerado en ciertos ciclos (1-4semanas), lo cual se le denomina Sprint.

Figura 4 *Equipo de desarrollo de la Metodología Scrum*



Fuente: (Lopez , 2015)

Para proporcionar un producto terminado (Sprint), un conjunto de profesionales considerado como equipo de desarrollo cumplen sus funciones en sus respectivas áreas de trabajo, el equipo es desarrollado de forma integral, otorgando ciertas diversidades de funciones de acuerdo a sus competencias y son auto-dirigidos, para la entrega cuentan con cierta libertad de decisión.

Scrum Master, cumple con la función de asegurarse que “el equipo” adopte las buenas prácticas, teorías y reglas de la metodología SCRUM, brindando al equipo del cual está a su servicio ayuda que los clientes externos puedan comprender las interacciones que les pueda ser de ayuda o no.

Product Owner, para el equipo de desarrollo, es el encargado de transmitirles la visión del producto que se anhela desarrollar, contribuyendo una expectativa de negocio.

Stakeholders, se refiere a aquellos individuos que no se constituyen parte directa del proceso de desarrollo del proyecto, sino que muestran también interés en el proyecto

como los directores, gerentes, comerciales, etc., de tal forma que también son consideradas.

Usuarios, de la misma manera que el anterior, no son parte del proceso de creación del proyecto, pueden estar presentes en la etapa de revisión de entregables siendo los destinatarios finales del proyecto a desarrollar, con el mismo objetivo.

Equipo de desarrollo, este equipo se organiza de una manera ordenada en la que cada parte del mismo es auto gestionable, y asimismo organiza la ejecución de los diferentes Sprints. Lo cual esto conlleva al responsable de hacer la entrega del producto final.

- **Extreme Programming (XP)**

“La programación extrema” es un método basado en una serie principios y reglas que se han empleado durante todo el asunto de desarrollo del software, haciendo el uso de cada de ellas para crear un proceso ágil. Para enfatizar la tarea de agregar valor y eliminar los procedimientos burocráticos.

Figura 5 Marco de trabajo de la Metodología XP



Fuente: (López, 2015)

La programación extrema se caracteriza por 12 principios, los cuales se encuentran agrupados en 4 categorías, dentro de cuales tenemos:

- **Retroalimentación a Escala Fina**, etapa donde encontramos diferentes principios como de planificación, de programación en parejas, realización de pruebas y cliente en sitio.
- **Proceso Continuo en lugar de por lotes**, la refactorización nos permite valorar el “diseño del sistema” a lo largo del tiempo en el cual ha sido desarrollado el proyecto y la codificación para hacer una pequeña cantidad de entrega, permitiéndonos una integración continua.
- **Entendimiento compartido**, en esta categoría definimos “la creación de la metáfora del sistema cierto criterios como de crear un diseño, entendible y fácil, tarjetas CRC (Clase, Responsabilidad y Colaboración).
- **Bienestar del programador**, esto conlleva a la creencia que un desarrollador exhausto, debería tener al menos 40 horas de trabajo a la semana y pocas horas extras de trabajo, para que así evite crear código de mala calidad.

- **Crystal Clear**

La metodología Crystal se establece el código de color para su definición de complejidad, si el color es más oscuro, significa que el método es más pesado, por lo que sí es más crítico el sistema se necesita ser más riguroso. Además, Crystal recomienda definir el color en cada elemento dependiendo de su criticidad y tamaño, en general no existe un método Crystal definido por cada tipo de proyecto en el que se desarrolla.

Figura 6 Citricidad de la metodología Crystal

L6	L20	L40	L80
E6	E20	E40	E80
D6	D20	D40	D80
C6	C20	C40	C80
Clear	Yellow	Orange	Red

Fuente (Lopez, 2015)

Cada letra muestra una referencia a los riesgos potenciales:

C: pérdida de confort debido a un fallo del sistema.

D: pérdida de dinero discrecional (dinero propio).

E: pérdida de dinero esencial (dinero del cual no se dispone).

L: pérdida de vidas por el fallo del sistema.

A su vez la cantidad de personas que son coordinadas en el proyecto, son indicadas por dichos números:

- Clear es para equipos de 8 personas o menos.
- Amarillo para equipos de 10 -20 personas.
- Naranja para equipos de 20-50 personas.
- Rojo para equipos de 50-100 y así sucesivamente pasando por el marrón y el violeta.

Como las metodologías descritas con anterioridad Crystal es regida por principios de hacen se utilización eficiente, entre los cuales se mencionan:

- Cada proyecto será necesario un grado diferente de compensación (Trade Off).

- Mientras menos sea la contextura del proyecto es mejor la forma de coordinación.
- Para cada proyecto serán necesarios distintos medios de comunicación.
- Para hacer la entrega a tiempo y sin errores a lo que el cliente está solicitando, los putos dulces hacen que el proyecto acelere en el desarrollo contando con personal capaz y dedicado a su trabajo.
- **Roles y Productos en Crystal Clear**

Se mencionan en Crystal Clear ocho distintos roles que son asignados acá cada miembro del equipo (writer, coordinador, tester, usuario experto, diseñador-programador, diseñador líder, sponsor ejecutivo, experto del negocio), los cuatro primeros roles deben estar asignados a personas dentro del mismo proyecto, mientras que los cuatro últimos están asignadas a personas diferentes.

Tabla 1 Roles y Productos en Crystal

Roll-Ultimo Responsable	Productos
Sponsor (patrocinador quien financia)	La declaración de la Misión con el Trade-off de Prioridades.
Equipo	La estructura y las convenciones de los equipos. Los resultados del trabajo de reflexión.
Coordinador, con ayuda del equipo	El mapa del Proyecto, El Plan de Entrega, El Estado del Proyecto, La lista de Riesgo, El plan y Estado de la Iteración, La visualización del Calendario-Cronograma.

Experto del negocio y usuario experto juntos	La lista de los objetivos por actor. Los Casos de Uso. El archivo de Requerimientos. El modelo de rol de usuario.
Líder del negocio (diseñador líder)	La descripción de la Arquitectura.
Diseñadores-programadores (incluyendo al líder de diseño)	Borradores de pantalla, Modelo de Dominio Común, Esquemas y notas de diseño, Código fuente, Código de Migración, Las pruebas, El sistema empaquetado.
Tester	Reporte de errores en ese momento.

Fuente: (López, 2015)

1.7.- Plataformas y Arquitecturas

Con la variedad de metodologías ágiles de desarrollo de software, también cuentan con diferentes plataformas con las cuales se puedan ejecutar, siendo las más usadas:

OpenProject: existen ciertos plugins para la gestión de la metodología ágiles como SCRUM, que a su vez incluye la gestión de timeline, workpackages, gestión documental. Wikis, Foros y News.

IceScrum: esta herramienta cubre la mayor parte de las prácticas ágiles, que está enfocada en soportar la metodología Scrum, Product Management, Sprints, Release

Plan, Timeline, Sandbox o Cuadros de mandos con una versión que puede ser gratuita o una versión extendida de pago.

TeamWork Project: esta herramienta virtual nos proporciona una gestión para los distintos proyectos y equipos de trabajo, ofrece un sistema de gestión y planificación de recursos, scheduling, ticketing (issue tracking), y facilidades para las metodologías ágiles como: Kanban y Scrum,

X Planner: X-Man (Extreme Manager) al trabajar con un formato ágil ya sea XP o Scrum, nos otorga una fácil gestión y seguimiento de proyectos ágiles, porque además es considera un programa limpio.

Agile Mantis: para desarrollar la metodología Scrum, podemos utilizar este plugin que viene en el programa MantisBT, el cual contiene todos los componentes necesarios para el desarrollo de algunos productos pendientes, historias de usuario, sprint, etc.

1.7.1. Las tecnologías más usadas en la investigación

1.7.1.1-Arduino

Según Ganazhapa (2016) Arduino es una plataforma de hardware de código abierto, basado en una placa con entradas y salidas, analógicas y digitales, que se desarrolla en un entorno de lenguaje de programación processing. Es un dispositivo que conecta al mundo físico con el mundo virtual, o también al mundo analógico con el digital.

Esta plataforma puede tomar información del entorno físico a través de sus puertas de entrada, para esto se utiliza una gama de sensores que se pueden usar para el control de luces, motores, pantallas y otras herramientas, creando una interfaz de comunicación de

un sistema a otro. El microcontrolador de dicha placa es programado mediante el lenguaje de programación Arduino (basado en wiring).

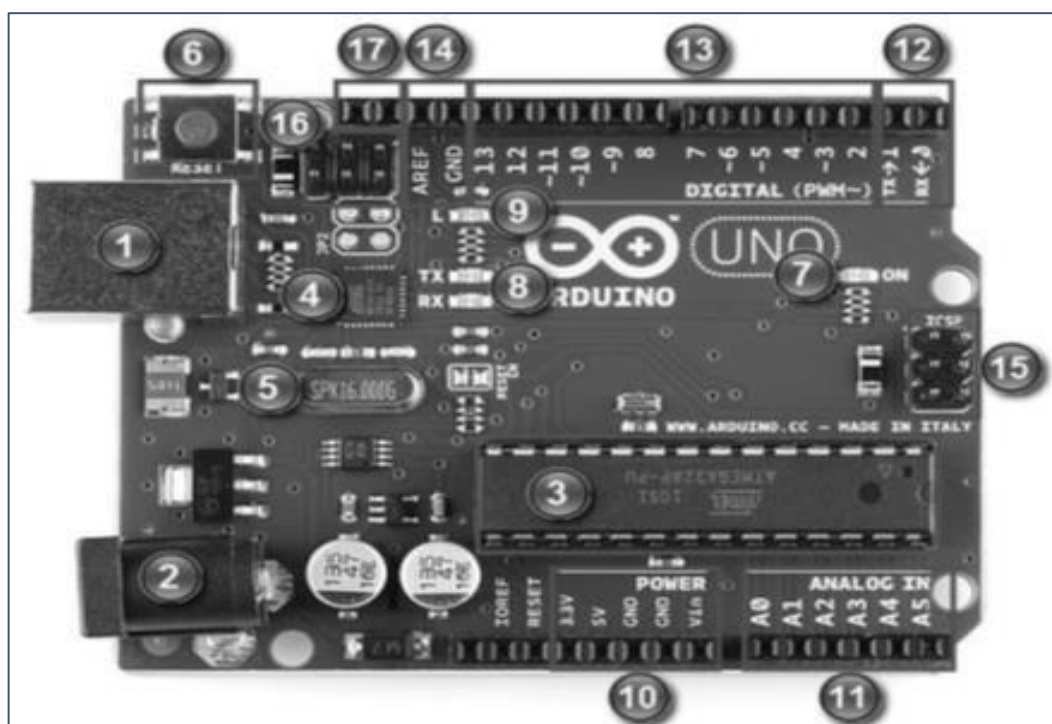
Las placas de Arduino pueden ser construidas a mano o montadas de fábrica. El software se puede descargar de manera manual y dichos ficheros de diseño están disponibles bajo una licencia abierta.

- **Placa Arduino y sus partes**

Una placa de hardware libre que incorpora un microcontrolador reprogramable y una serie de pines-hembra (los cuales están unidos internamente a las patillas de E/S del microcontrolador) que permiten conectar allí de forma muy sencilla y cómoda diferentes sensores y actuadores. Cuando hablamos de “placa hardware” nos estamos refiriendo en concreto a una PCB (del inglés “printed circuit board”, o sea placa de circuito impreso). Las PCBs son superficies fabricadas de un material no conductor (normalmente resinas de fibras de vidrios reforzadas, cerámica o plástico). Así pues, la placa Arduino no es más que una PCB que implementa un determinado diseño de circuitería interna. (Torrente, 2013).

Según Ganazhapa (2016), nos afirma que el Arduino UNO es una placa electrónica basada en el ATmega328P. Cuenta con 14 puertos digitales de entrada/salida (de los cuales 6 se pueden utilizar como puertos de salida PWM), 6 entradas analógicas, un resonador de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSO y un botón de reinicio. Contiene todo lo necesario para desarrollar una infinidad de proyectos prácticos de electrónica; simplemente conectarlo a un ordenador con cable USB o con un adaptador de corriente continua para empezar.

Figura 7 Placa Arduino UNO



Fuente: (Arduino, 2016)

Tabla 2 Partes de la placa Arduino UNO

Nº	DESCRIPCION DE PARTES
1	USB para el cable tipo AB
2	Conector de voltaje de 9 a 12 voltios
3	Microcontrolador ATmega 328p, cerebro de Arduino
4	ATmega 16U2 encargado de la comunicación USB con el PC
5	Cristal de cuarzo de 16 MHz
6	Pulsador de Reset
7	LED verde de placa encendida
8	LED TX (transmisión) y LED RX (Receptor) de la comunicación serial
9	LED naranja conectado al pin 13
10	Puertos de voltaje 5 voltios, 3,3 voltios y tierra
11	Entradas analógicas
12	Puertos de recepción (RX=0) y transmisión (TX=1) serial
13	Puertos de E/S digitales y PWM
14	Puertos de referencias analógicas y tierra
15	Puertos ICSP para programación serial
16	Puertos ICSP para interfaz USB
17	Puertos I2C (SDA,SCL)

Tipos de Arduino

- **Arduino Uno:** Es una placa de Microcontrolador basado en ATmega328P (hojas de datos). Tiene 14 pines digitales de entrada/salida (de las cuales 6 se pueden usar como salidas PWM), 6 entradas analógicas, un cristal de cuarzo de 16 MHZ, una conexión USB., un conector de alimentación.
- **Arduino Leonardo:** Es un tablero de microcontrolador basado en ATmega 32u4 (hoja de datos), tiene 20 pines digitales de entrada y salida (de las cuales 7 pueden usarse como salidas PWM y 12 como entradas analógicas), un oscilador de cristal de 16 MHZ, una conexión micro USB, un conector de alimentación, un encabezado ICSP y un botón de reinicio.
- **Arduino 101:** Mantiene el mismo factor de forma robusta y la lista de periféricos de UNO con la incorporación de las capacidades de Bluetooth LE a bordo y un acelerómetro giroscópico de 6 ejes.
- **Arduino Esplora:** Tiene salida de sonido y luz a bordo, y varios sensores de entrada, incluyendo un joystick, un control deslizante, un sensor de temperatura, un acelerómetro, un micrófono y un sensor de luz.
- **Arduino Mega:** El MEGA 2560 está diseñado para proyectos más complejos. Con 54 pines de entrada y salida, 16 entradas analógicas y un espacio más grande para su boceto. Es la placa recomendada para impresoras 3D y proyectos de robótica.
- **Arduino Yún:** El Yún se distingue de otras placas Arduino por su capacidad para comunicarse con la distribución Linux a bordo, dando como resultado una poderosa computadora en red con la finalidad de un Arduino.

1.7.1.2. Microcontroladores

Según Torrente (2013) un microcontrolador es un circuito integrado o “chip”(es decir, un dispositivo electrónico que integra en un solo encapsulado un gran número de componentes) que tiene la característica de ser programable. Es decir que es capaz de ejecutar de forma autónoma una serie de instrucciones previamente definidas por nosotros. Un microcontrolador incluye tres elementos básicos:

- **CPU (Unidad Central de Proceso):** Es la parte encargada de ejecutar cada instrucción y de controlar dicha ejecución que se realice correctamente. Dichos datos disponibles son los de entrada y salida.
- **Diferentes Tipos de Memorias:** encargadas de alojar tanto las instrucciones como los diferentes datos que estas necesitan. De esta manera posibilitan que toda esta información (instrucciones y datos) este siempre disponible para que la CPU pueda acceder y trabajar con ella en cualquier momento.
- **Diferentes Patillas de E/S (Entrada y Salida):** Son las encargadas de comunicar el microcontrolador con el exterior. En las patillas de entrada del microcontrolador podemos conectar sensores para que este pueda recibir datos provenientes de su entorno y en sus patillas de salida podremos conectar actuadores para que el microcontrolador pueda enviar órdenes e así interactuar con el medio físico.

1.7.1.3. Sensores

Según Crespo E. (2014) los sensores son el primer eslabón de la cadena de IoT, en su forma más abstracta, un sensor no es más que una caja negra que transduce un estado físico que queremos monitorear en una propiedad eléctrica que podemos medir. Puede tan simple como

un conjunto de contactos que cambian entre abierto y cortocircuito cuando pulsamos un botón o abrimos una ventana. Puede ser un potenciómetro que cambia la resistencia cuando un humano gira un dial o un robot extiende un brazo. También puede ser tan complicado como un circuito integrado que detecta la aceleración y el magnetismo a lo largo de tres ejes para determinar la orientación, la aceleración y el rumbo de la brújula hacia su dispositivo IoT.

Los sensores se pueden clasificar en función de los datos de salida en:

- **Digitales:** en los sensores digitales, estos nos pueden dar una señal digital simple con dos estados como una salida de contacto libre de tensión o una salida en bus digital. En caso que el sensor use comunicación por bus, deberemos usar algunos de los buses que implementa Arduino o usar Hardware adicional que nos proporcione una interfaz entre el Arduino y el bus.
- **Analógicos:** como ejemplo tomamos el sensor ACS714, es un sensor de efecto hall que mide las corrientes eléctricas que pasan a traves del chip y devuelve un valor en voltaje proporcional a la corriente que circula por el sensor.

Características de los sensores

- Rango de medida: dominio de la magnitud medida en el que puede aplicarse el sensor.
- Precisión: es el error de medida máximo esperado.
- Offset o desviación de cero: valor de la variable de salida cuando la variable de entrada es nula. Si el rango de medida no llega a valores nulos de la variable de entrada.
- Linealidad o correlación lineal

- Sensibilidad de un sensor: suponiendo que es de entrada a salida y la variación de la magnitud de entrada.
- Resolución: mínima variación de la magnitud de entrada que puede detectarse a la salida.
- Rapidez de respuesta: puede ser en un tiempo fijo o depender de cuanto varié la magnitud a medir. Depende de la capacidad del sistema para seguir las variaciones de la magnitud de entrada.
- Derivas: son otras magnitudes, aparte de la medida como magnitud de entrada, que influyen en la variable de salida. Como la humedad, la temperatura, etc.
- Repetitividad: error esperado al repetir varias veces la misma medida.

1.7.1.4.- Tipos de Sensores

Para la realización de este proyecto se tomarán en cuenta diferentes tipos de sensores:

- **DHT22 Sensor digital de temperatura y humedad**

Utiliza un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos mediante una señal digital en el pin de datos (no hay pines de entrada analógica). Es bastante simple de usar, pero requiere sincronización cuidadosa para tomar datos. El único inconveniente de este sensor es que sólo se puede obtener nuevos datos una vez cada 2 segundos, así que las lecturas que se pueden realizar serán mínimas cada 2 segundos. Sólo tienes que conectar el primer pin de la izquierda a la fuente de alimentación 3 -5V, el segundo pin a tu pin de entrada de datos y el cuarto (último) pin a tierra. ¡A pesar de que utiliza un solo cable para enviar datos no es compatible con el protocolo de Dallas One-Wire! Si desea conectar varios sensores, cada uno debe tener su propio pin de datos, Hay una librería de ejemplo disponible para Arduino.

Características

- Resolución Temperatura: 0.1°C
- Alimentación: $3.3\text{ Vdc} \leq V_{cc} \leq 6\text{Vdc}$
- Precisión de medición de temperatura: $< \pm 0.5^{\circ}\text{C}$
- Rango de medición de temperatura: -40°C a 80°C
- Precisión de medición de humedad: 2% RH
- Rango de medición de humedad: De 0 a 100% RH
- Resolución de Humedad: 0.1% RH
- Tiempo de sensado: 2s

Figura 8 DTH22, sensor de temperatura y humedad



Fuente: (Perea, 2016)

- Sensor de humedad de suelo SKU

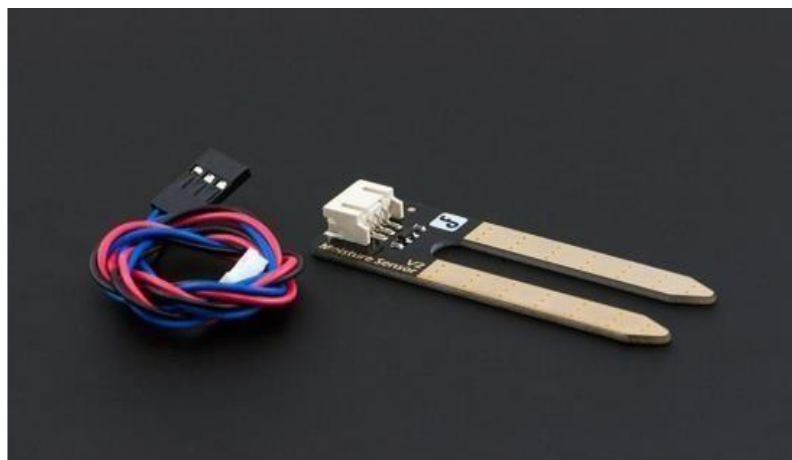
Este sensor puede leer la cantidad de humedad presente en el suelo que lo rodea. Es un sensor de baja tecnología, pero es ideal para el seguimiento de un jardín urbano o un cultivo. Se trata de una herramienta indispensable para saber el estado de humedad del suelo. Cuenta con dos sondas para pasar corriente eléctrica a través del suelo

circundante, y luego lee la resistencia para obtener el nivel de humedad. Más agua hace que el suelo conduzca la electricidad con mayor facilidad (menos resistencia), mientras que el suelo seco es un mal conductor de la electricidad (más resistencia). Será útil para recordarle que debe regar las plantas o para monitorear la humedad del suelo en su jardín o cultivo.

Especificaciones

- Fuente de alimentación: 3,3 V o 5 V
- Señal de tensión de salida: 0 ~ 4.2v
- Corriente: 35mA
- Definición de los pines:
 - Salida analógica (cable azul)
 - GND (alambre negro)
 - Alimentación (cable rojo)
- Tamaño: 60x20x5mm (2.36x.0.79x0.2")
- Acabado superficial: oro de inmersión

Figura 9 Higrómetro, módulo SKU



Fuente: (Perea, 2016)

- **Sensor Luminosidad**

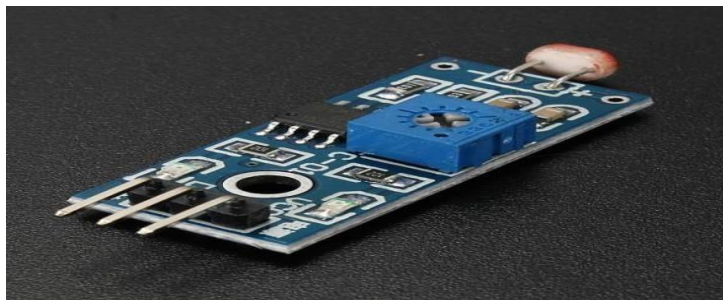
Un “LDR” viene a ser un dispositivo en donde su resistencia va a variar de acuerdo con la cantidad de luz recibida. Estos dispositivos son bastante utilizados en proyectos como los de control de iluminación, interruptores crepusculares seguidores solares, etc.

La salida digital suele utilizarse para controlar un relay y realizar una acción dependiente de la intensidad de luz, esta conserva solo dos estados: “activo/apagado”, por lo que el cambio de un estado a otro va a depender del umbral que se fije con el potenciómetro del módulo.

Características:

- Detecta la luminosidad de la intemperie y la intensidad de la luz
- Su sensibilidad es ajustable “mediante potenciómetro digital azul”
- Indicador de encendido “rojo”
- Agujero de perno fijo que permite una fácil instalación

Figura 10 *LDR (sensor de luminosidad)*



Fuente (Perea, 2016)

Sensor de lluvia

Este tipo de sensor detecta la presencia de lluvia por la diferenciación de conductividad del sensor al entrar en contacto con el agua. Existen varios modelos de sensores similares, como el FC-37 y el YL-83. (Tostratonic,2013)

Características

- Tamaño de PBC: 3.2cm x 1.4cm
- Voltaje de funcionamiento: 3.3V ~ 5V
- Tamaño de celda: 5cm x 4cm

Figura 11 *Sensor de lluvia*



Fuente: (Tostratonic, 2013)

Módulo Bluetooth HC - 06

Según MECHATRONICS, (2013) este módulo nos permite conectar nuestro proyecto Arduino de forma inalámbrica (Bluetooth) a un Smartphone, teléfono móvil o PC y operarlo fácilmente a través del puerto serie. La transmisión es completamente transparente para el programador, por lo que se vincula directamente al pin serial de nuestro microcontrolador preferido (considerando el nivel de voltaje, ya que el módulo se alimenta con 3.3V).

Figura 12 *Modulo Bluetooth HC – 06*



Fuente: (Mechatronics, 2013)

1.7.1.5. Actuadores

- **Bomba de liquido**

Según (Mechatronics, 2013), este tipo de bomba se puede controlar empleando un driver de potencia y un microcontrolador. El empleo del driver es necesario puesto que la corriente y el voltaje de la bomba es mayor al usado por el microcontrolador. La potencia de este motor es de “19W” por lo que permite un flujo mediano de agua de hasta “800 litros por hora”.

Los microcontroladores que podemos usar son Arduino, Pic, Raspberry Pi o un Módulo ESP8266(Wemos, NodeMCU)

Características

- Voltaje de operación: 5 – 12V DC
- Temperatura del fluido: menor a 100°C
- Tamaño: 8x8x6.5cm
- Peso: 220gr.

Figura 13 *Bomba de liquido*



Fuente (Mechatronics, 2013)

- **Electroválvulas**

Según NOVAGR (2016), son válvulas hidráulicas que adicionadas mediante un sistema eléctrico y conectadas a un programador de riego cierran o abren el paso de agua en una conducción. Una electroválvula es el resultado una válvula hidráulica mediante un sistema eléctrico se abre o cierra. De esta manera se simplifica la red de riego, automatizando la entrada y clausura de las válvulas de la instalación a través de impulsos eléctricos originador por un programador de riego.

Figura 14 *Electroválvula*



Fuente (NOVAGR, 2016)

- **Relay**

(Mechatronics, 2013) con Arduino se puede realizar bastantes proyectos, con éste se puede querer controlar componentes de voltaje alto, como bombas o bombillas de agua, sin embargo, no puede ser esto posible utilizando solo con Arduino, por lo que es inevitable emplear Relays o Reles, que son dispositivos electromagnéticos compuestos de un núcleo ferromagnético fijo y otro móvil, ya que esto permite el control del “encendido y el apagado” de cualquier aparato que se conecte a la fuente de alimentación eléctrica.

Dentro de las cargas que se pueden manejar son: motores AC (220V), electroválvulas, bombillas de luz, luminarias, calentadores de agua, motores DC, solenoides, etc.

Figura 15 *Relay*



Fuente (Mechatronics, 2013)

1.7.1.6. Periféricos

- **Display**

(Mechatronics, 2013) nos dice que los sistemas digitales empotrados como Arduino, Pic u otros incrustados trabajan exclusivamente con lógica binaria (0 y 1), por lo que para “leer los datos” empleamos pantallas o displays alfanuméricas LCD. Estas pantallas

pueden servir para realizar “debugging o correcciones” en tus distintos proyectos, más que todo en el manejo de sensores y procesamientos de datos.

Posee 4 filas y 20 columnas de dígitos alfanuméricos, funciona con el controlador interno HD44780, un integrado utilizado mucho con una amplia documentación existente sobre ello. Para realizar la conexión de la pantalla LCD a nuestro Arduino/PIC es necesario seis pines: 2 pines de control y de datos. Con respecto a la programación en Arduino está incluida por defecto la librería Liquid Crystal, el cual contiene ejemplos de prueba.

Características:

- Interface de comunicación: paralelo 4 u 8 bits
- Voltaje de Operación_ 5V
- Color Texto: Blanco
- Backligth: Azul

Figura 16 *Display*



Fuente: (Mechatronics, 2013).

- **Diodo LED**

Para Castillo (2017), es un componente semiconductor el cual tiene como propiedad el de emitir luz cuando es travesado por una corriente en polarización directa. De la misma forma que los otros tipos de diodos, “los LEDs” tienen dos patillas de conexión

(un ánodo y un cátodo). Cuando el “diodo” es nuevo, el terminal largo es ánodo (+) y el corto el cátodo (-). Asimismo, se puede diferenciar observando el LED al trasluz. “El cátodo” es una pieza con un tamaño mayor que “el ánodo”, además posee el cuerpo redondeado en cambio el cátodo es plano.

Figura 17 *Diodo LED*



Fuente: (Castillo, 2017)

- **Microprocesador**

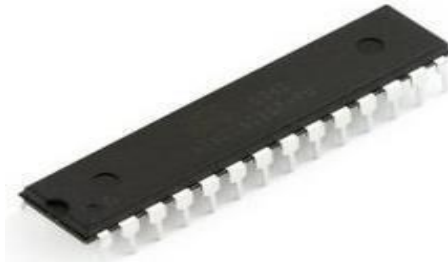
Un microprocesador es un “chip” o sistema integrado electrónico que permite interpretar y ejecutar órdenes. La principal función es la de realizar cálculos y operaciones (aritméticas y lógicas), sobre los datos que son enviados por los programas, dirigiendo y controlando todo el proceso de datos (Villar, 2017).

Consiste en unas pastillas de conexión insertada en una placa base del ordenador, esta base está compuesta de cristal plástico y silicio, equivalente a millones de componentes electrónicos. El microprocesador accede a la información, la procesa y da la respuesta adecuada. Está dividida en tres partes, de las cuales cada una tiene una tarea fijada:

- **Unidad de control:** se encarga de interpretar las instrucciones que le llegan de la memoria RAM.
- **Unidad aritmética lógica:** realiza operaciones lógicas y matemáticas que le solicita la Unidad de Control.

- **Registros de almacenamiento:** permite almacenar los resultados de los cálculos realizado por la unidad aritmética lógica, además de dirección de memoria.

Figura 18 *Microprocesador*



Fuente (Villar, 2017)

1.7.2. Lenguajes de programación

En esta sección veremos los lenguajes de programación que usaremos en nuestro sistema inteligente, para el desarrollo de la aplicación móvil como sus IDE donde se desarrollaran:

1.7.2.1.-Lenguaje Arduino

Una placa Arduino puede leer una entrada (luz en un sensor, dedo en un botón o un tweet) y convertirla en una salida: encender un motor, encender un LED, publicar algo en línea. Le dices a tu placa qué hacer enviando un conjunto de instrucciones al microcontrolador de la placa.

Pomares (2009), afirmó que el entorno del lenguaje de programación Arduino es fácil de usar y lo suficientemente flexible para los usuarios. Arduino se basa en un entorno de programación de procesamiento que permite el desarrollo basado en C++.

1.7.2.2.-Lenguaje C++

Según Crespo (2014), C++ es un tipo de lenguaje de programación diseñado por Bjarne Stroustrup a mediados de los años 80.

El C++ es un tipo de lenguaje híbrido, desde la perspectiva de los lenguajes orientados a objetos. Consecutivamente, se añadió a la programación dos paradigmas que ya estaban admitidos, uno de ellos es la programación estructurada y el otro es la programación orientada a objetos, razón por la cual el C++ se considera un lenguaje de programación multiparadigma.

La Programación Orientada a Objetos es un paradigma que nos refleja una mejora en la forma de poder obtener mejores resultados al codificar. Los objetos que se manipulan son aquellos datos de entrada para poder así conseguir datos de salida, en donde cada objeto brinda una especial y mejor funcionalidad.

1.7.2.3.-Java

Java fue creado para dispositivos electrónicos relativamente pequeños como la calculadora, la televisión, el refrigerador y el microondas. Consiguientemente con la aparición de electrodomésticos de gran variedad y los constantes cambios en los mismos, era indispensable una herramienta que no dependa del tipo aparato, por lo que era necesario un código neutro; la principal idea fue ejecutar los códigos en una máquina virtual que lo creara portable convirtiéndolo el código en uso particular por el electrodoméstico. (Renon, 2013)

Java se relaciona con C++, ya que es un derivado directo de C. la mayor parte del carácter de java esta heredado de estos dos lenguajes. De C, Java deriva su sintaxis, la totalidad de las características orientadas a objetos están basadas en C++.

1.7.2.4.-Php

Cobo (2005) afirma que el lenguaje es un "lenguaje interpretado de lado del servidor que surge dentro de la corriente denominada código abierto" (open source). El autor también destaca su versatilidad, potencia, modularidad y robustez. Asimismo, menciona que los programas son integrados directamente dentro del código HTML.

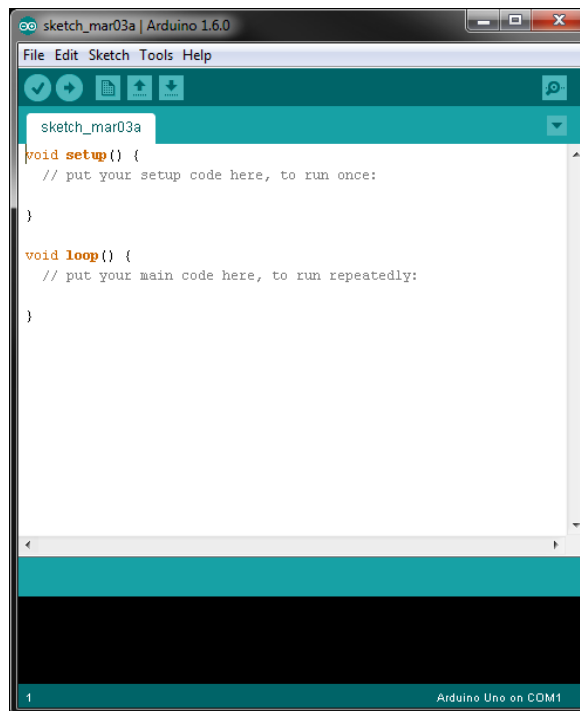
1.7.3.- Entorno a la programación

1.7.3.1.1.- IDE de Arduino

El entorno para el desarrollo en Arduino (IDE, Integrated development environment). Según Diaz (2014), se encuentra conformado por un editor de texto el cuál es útil para trazar el código, un área de mensajes, una barra de herramientas con botones para las funciones comunes, una consola de texto, y una serie de menús. Permite conexión por USB con el hardware Arduino, donde se puede cargar los programas y facilita el inicio de comunicación con ellos, para escribir el código fuente o programa de aplicación lo que denomina “sketch” (programa).

Para cargar los programas se necesita que Arduino esté conectado al ordenador mediante el USB, lo cual nos permitirá observar el tiempo de ejecución de datos del programa mediante la consola serie.

Figura 19 IDE de Arduino



Fuente: (Crespo , 2014)

1.7.3.1.2.- Android Studio

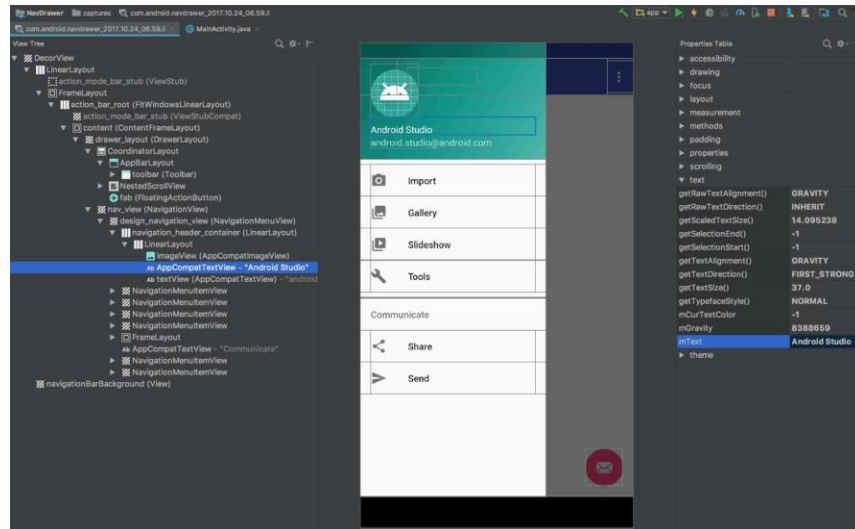
Es el entorno de desarrollo integrado que se presentó en el Google I/O en 2013 y pasó a ser la herramienta oficial para el desarrollo de aplicaciones, este IDE fue desarrollado por Google y está basado en IntelliJ, por lo que existe diferencias notables a comparación con Eclipse, esto no complica ni facilita el desarrollo de una aplicación, sino que cambian algunas características como la compilación, uso de librerías por paquetes externos, pero el código de una aplicación funciona de la misma forma sin importar el IDE que se utilice (Luján 2019).

En el desarrollo en Android se debe entender que se necesitan 3 partes importantes:

- Java “lenguaje de programación para programar en Android”
- Entorno de desarrollo “Herramienta para desarrollar aplicaciones”

- Android “Librerías de Android”

Figura 20 *Android Studio*



Fuente: (Developers, 2021)

1.7.4 Sistema de Gestión de Base de Datos MySQL

Natsys (2014), se refiere a un Sistema de Gestión de Base de Datos relacional, multihilo y multiusuario creado por la empresa MySQL AB ideal para este tipo de aplicaciones. Redundancia de datos: ocurre cuando reiteradamente se almacena la misma información varias veces en una misma base de datos, este suceso debe evitarse, ya que dificulta la tarea de modificación de datos y es el motivo más frecuente de inconsistencia de datos, además que requiere mayor espacio de almacenamiento.

- Inconsistencia de datos: “puede darse el caso que dos aplicaciones diferentes proporcionen resultados distintos para el mismo dato”, es decir, en que no todas las copias redundantes contienen la misma información, así mismo esas formas pueden conducir a datos almacenados en distintos sitios.

- Integridad de datos: usualmente los datos son repartidos en varios archivos. Como pasa con “MySQL”, en donde la base de datos siempre está a la disponibilidad para varios usuarios de manera simultánea, cabe recalcar que deben existir mecanismos que aseguren que las relaciones entre registros se mantengan coherentes, y respeten las dependencias de existencia y que las claves

1.7.5.-El Riego Tecnificado por Goteo

Según Environment (2015), se refiere a gestionar el eficiente de los recursos acuíferos, por medio del uso adecuado de la tecnología en favor de la agricultura; está diseñado para saber cuándo, cuanto y como regar, admitiendo la aplicación en los cultivos de agua, fertilizantes y nutrientes de manera segura.

Beneficios del riego tecnificado

La productividad del riego tecnificado podemos expresarlo en la cantidad de productos agrícolas derivados por cada parcela donde se ha aplicado el agua, el cual varia por los diferentes tipos de cultivos, clima, los valores de las producciones obtenidas como son:

- Mayor eficacia en el uso del agua y los fertilizantes.
- Mayor producción y mejor calidad de los productos; por tanto, más ganancias.
- Mayores ingresos económicos para los agricultores.

Riego por Goteo

Según Medina (1985), nos comenta que el riego por goteo es una nueva tecnología para la aplicación del agua de riego, que se está extendiendo rápidamente a nivel mundial y está experimentando un proceso de rápida mejora. La técnica consiste en entregar agua al campo a

través de tuberías de plástico de pequeño calibre, que se colocan a lo largo de las hileras de plantas, entregando agua de manera lenta pero frecuente al sistema de raíces a través de dispositivos apropiados llamados goteros o goteros.

Tabla 3 *Ventajas y desventajas del Riego por goteo*

VENTAJAS	DESVENTAJAS
Permite un crecimiento adecuado del sistema de raíces.	El sistema de goteo puede taparse si no se filtra el agua correctamente.
Disminuye el gasto del agua y la eficiencia del uso del agua se incrementa notablemente.	La inversión inicial es alta y es indispensable contar con personal técnico capacitado para el diseño.
Se adapta a terrenos rocosos o con pendientes.	Los costos de instalación y diseño son elevados.

Fuente (Agroware, 2016)

(Agroware, 2016) nos dice que, el riego por goteo evita las fluctuaciones de humedad que se tiene con los otros tipos de riego. Puede regarse 2 o 3 veces por día. Se puede utilizar tanto a cielo abierto como en cultivos bajo invernadero.

Figura 21 *Riego por goteo*



Fuente: (Agroware, 2016)

Tipo de suelos

Según Bellver (2018), entre los distintos tipos de suelo tenemos los arenosos, los limosos o los de turba. Conocer las peculiaridades de cada tipo de suelo es importante en la agricultura, ya que según el tipo se pondrán hacer cultivos o proyectos, hay básicamente tipos de suelo las cuales son:

- Suelos Arenosos

Los suelos arenosos contienen partículas más grandes que otros suelos. Se siente áspero y seco al tacto porque las partículas que lo componen están muy separadas entre sí y no retienen bien la humedad. En suelos arenosos, no son aptos para la agricultura porque no pueden retener los nutrientes. Debido a la velocidad a la que se pierde agua, las plantas en suelos arenosos no tienen la oportunidad de usar los nutrientes de manera eficiente.

Figura 22 Suelo arenoso



- **Suelos Calizos**

Llamamos suelos de piedra caliza a las pequeñas rocas blancas naturales. En su composición encontramos carbonato de calcio, magnesio y otros minerales como arcilla, cuarzo o hematites. Este es un suelo particularmente seco y muy árido. Además, dado que contiene carbono del calcio, se seca rápidamente y evita que obtenga los nutrientes del suelo de manera adecuada a través de la planta. Por eso no se recomienda plantar en suelo calizo, ya que no tiene humedad ni nutrientes y es difícil que las plantas sobrevivan.

Figura 23 *Suelos calizos*



- **Suelos Limosos**

Estos tipos de suelo están formados por juntas más pequeñas y son más suaves al tacto que los suelos arenosos y limosos que retienen el agua y los nutrientes durante más tiempo. Es de color marrón oscuro y el limo es una mezcla de arena fina y arcilla que, junto con la tierra y restos vegetales, forma una especie de lodo. Este tipo de suelo se encuentra a menudo en los lechos de los ríos. Son suelos muy fértiles considerando su humedad y nutrientes. Más fácil de cultivar que los suelos arenosos o arcillosos. Casi todos los tipos de árboles y plantas

pueden crecer en este tipo de suelo, excepto aquellos que requieren condiciones muy secas.

Figura 24 *Suelos Limosos*



- **Suelos Humíferos o de tierra negra**

Se llama suelo húmedo al suelo que ya tiene materia orgánica descompuesta.

En este tipo de suelo logramos distinguir microorganismos u organismos que obtienen ser muy favorables para las plantas. De esta forma, los humedales son los más utilizados para la mejora de actividades agrícolas.

Figura 25 *Suelos Humíferos*



- **Suelos Arcillosos**

Este suelo consiste en arcilla amarillenta de grano fino en un 45%, que retiene mucha agua y forma charcos, que son buenos para la labranza si se mezclan con humo. Este tipo de suelo tiene una gran capacidad para retener agua y nutrientes, pero tiene poca porosidad y es difícil de procesar. Su textura y pegajosidad hacen que las raíces no se aireen lo suficiente y se pudran. En este tipo de suelo no se pueden cultivar plantas ni árboles de ningún tipo.

Figura 26 *Suelos Arcillosos*



- **Suelos Pedregosos**

Este tipo de suelo se denomina así por las pequeñas estructuras de piedra en su composición. Esto sucede porque la superficie de la Tierra se despliega por causas naturales o inspiradas. El mayor problema de este tipo de suelo es que es semipermeable, por lo que no deja entrar el agua. De esta forma, cultivar en este tipo de suelo es muy complicado, aunque existe una xerófita, si puede

crecer en este tipo de suelo, algunos cultivos típicos en estos suelos son: milenrama, caléndula o lavanda.

Figura 27 Suelos Pedregosos.



- **Suelos de Turba**

Excelente suelo de cultivo, utilizado como sustrato de cultivo en la agricultura.

El suelo de turba es de color marrón oscuro o negro, de textura suave y alto en humedad y nutrientes. La turba generalmente está saturada con agua, pero una vez drenada es excelente para el cultivo. Una propiedad valiosa de la turba es su capacidad para retener agua en los meses secos y proteger las raíces de las bajas temperaturas. En invierno, los suelos de turba contienen un pH ácido entre 3,5 y 4. Los agricultores los utilizan para regular la química del suelo, así como los agentes de control de plagas del suelo. Los suelos de turba son ideales para una variedad de semilleros porque son porosos y proporcionan buena humedad.


Figura 28 *Suelo de turba*











Tipos de Cultivos

Dentro del fundo La Peña se encontrarán diversos tipos de cultivos de los cuales son los más frecuentes en su producción:

Tabla 4 *Tipos de cultivos*

NOMBRE	CARACATERISTICAS	CONDICIONES AMBIENTALES	IMAGEN
Alfalfa	Altura: 1m.	Temperatura: 2 – 3 °C Riego: moderado Ubicación: a plena luz	

Moringa	Altura: 10 a 12 mts.	Temperatura: 22 – 35 °C Riego: moderado Ubicación: luz directa	
Pasto elefante	Altura: 3m.	Temperatura: 18– 30 °C Riego: moderado Ubicación: a plena luz	
Pasto alemán	Altura: 2m.	Temperatura: 19 – 29 °C Riego: moderado Ubicación: a plena luz	
Maíz	Altura: 4m.	Temperatura: 20 – 32 °C Riego: moderado Ubicación: a plena luz	

Lima	Altura: 2 a 4mts.	Temperatura: 23 – 32 °C Riego: moderado Ubicación: a plena luz	
Naranja	Altura: 8m.	Temperatura: 23 – 32 °C Riego: moderado Ubicación: a plena luz	
Algodón	Altura: 60 a 150 cm.	Temperatura: 2 – 3 °C Riego: abundante Ubicación: luz indirecta	
Espárragos	Altura: 10 a 12 cm.	Temperatura: 18 – 25 °C Riego: moderado Ubicación: a plena luz	

Fuente: (EcuREd, 2012)

CAPÍTULO II: MÉTODOS Y MATERIALES

2.1.-Diseño de Contrastación de Hipótesis

La presente investigación es de tipo no experimental aplicada debido a que los datos e información son obtenidos de manera teórica, practica, para ello se recurrió a bibliografías establecidas en libros, publicaciones electrónicas, revistas y otros. Esto ayudará en la elaboración del proyecto de tesis de investigación, y estará elaborado por dos estudiantes de la escuela profesional de Ingeniería en computación e informática.

$$O_1 \Longrightarrow X \Longrightarrow O_2$$

Dónde:

O₁: Momento antes de desarrollar el prototipo en Arduino y la aplicación móvil.

X: Desarrollo del sistema informático

O₂: Momento después de desarrollar el prototipo en Arduino y la aplicación móvil.

- Definición y operacionalización de variables

Tabla 5 Variable Dependiente

Variable Dependiente	Definición	Dimensiones	Indicadores
Ahorro De Agua	Un sistema de riego se refiere al aprovechamiento eficiente de los recursos acuíferos, a partir del uso adecuado de la tecnología	Requisitos funcionales	Funcionamiento del apertura del agua
			Funcionamiento del cierre del agua

Fuente: Elaboración propia

Tabla 6 Variable Independiente

Variable Independiente	Definición	Dimensiones	Indicadores
Sistema Informático De Control y Monitoreo	Un Sistema Informático está compuesto por un conjunto de elementos que están orientados al procesamiento y administración de datos e información	Componentes del hardware	Arduino UNO Microcontroladores
		Sensores	Funcionamiento del temperatura y humedad. Funcionamiento del sensor humedad de suelo. Funcionamiento del Sensor de lluvia Funcionamiento del sensor Ultrasónico del nivel de agua. Funcionamiento del módulo bluetooth.
		Actuadores	Bomba de liquido Relay
		Periféricos	Display Diodo led Microprocesador

		Aplicación móvil	Lenguaje Java Android Studio MySQL
--	--	---------------------	--

2.2.-Poblacion y Muestra

- **Población**

Se tomará como población las diferentes variables Meteorológicas, las cuales son parámetros, que por medio de su comportamiento permitirá conocer la condición de la atmosfera en su momento.

Tabla 7 Población

Variables Meteorológicas
Temperatura Humedad Radiación
Luminosidad Precipitación
Nubes Viento Calidad de Aire Presibilidad

- **Muestra**

Se tomará como muestra las siguientes variables Meteorológicas

Tabla 8 Muestra

Variables Meteorológicas
Temperatura
Humedad
Luminosidad

2.2.-Técnicas e instrumentos de la recolección de datos

Técnicas: Observación.

Instrumentos:

Ficha De Registro De Datos

Cuestionario.

Materiales:

- **02 laptops**

Marca: HP Y LENOVO Sistema operativo: Windows 10

- **02 celulares**

Sistema operativo: Android 10.

- **01 ARDUINO UNO**

Cable USB 2.0 para Arduino.

Protoboard.

Sensor de Temperatura.

Sensor de Lluvia.

Sensor de Humedad.

Proteus.

CAPÍTULO III: RESULTADOS Y DISCUSIÓN

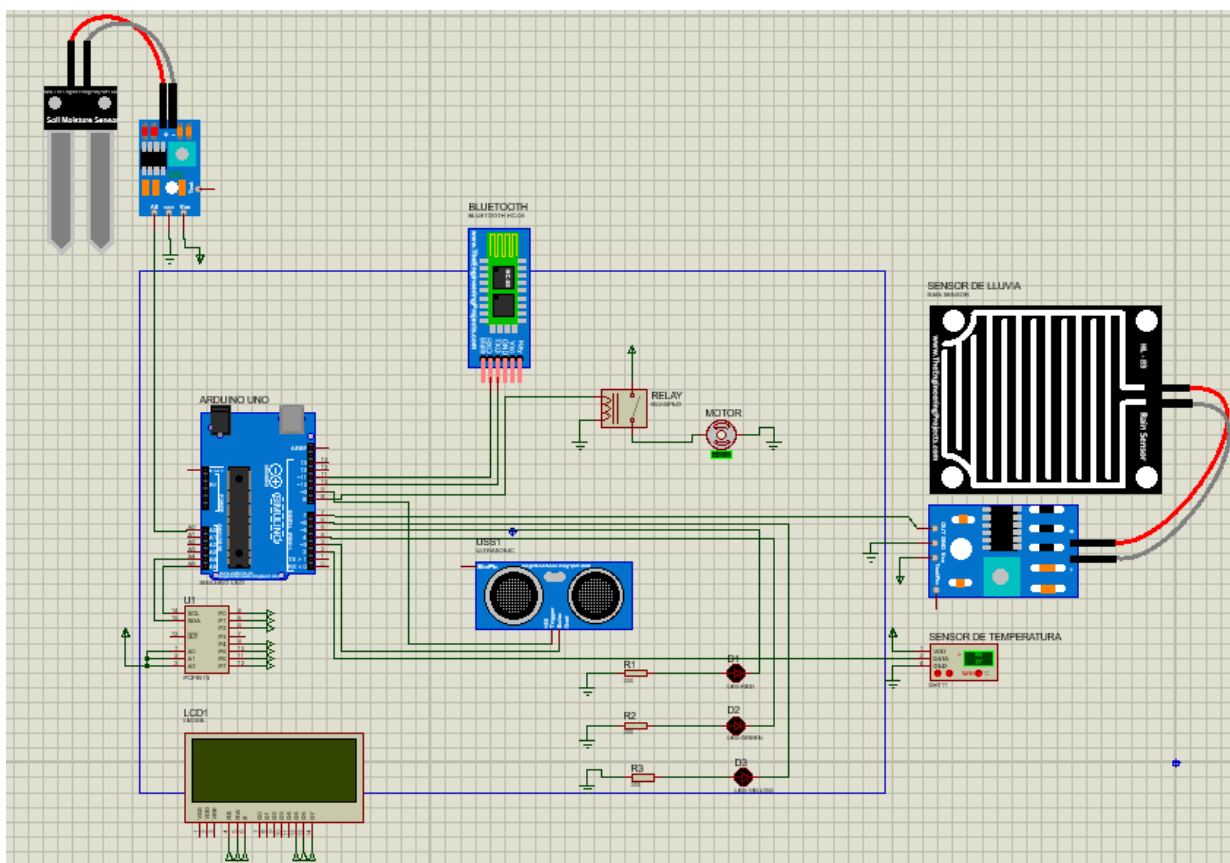
3.1.-Resultados

La situación actual de los cultivos en el fundo la peña se determina mediante la calidad de suelo, la cantidad de agua y el cambio climático, por lo que se entrevistó al encargado del fundo la peña de la Universidad Nacional Pedro Ruiz Gallo, en la que se mencionó y/o indico lo siguiente, que siendo el riego de manera tradicional en el fundo, lo cual hacen el uso de mangueras de gran longitud, estando estas conectadas a un tanque de agua, permitiendo abastecerse, para ello, también hacen uso de baldes, para poder alcanzar el riego en las parcelas más alejadas, ocasionando la perdida insuficiente de agua, por ende, se optó a implementar un sistema de automatización, con la finalidad de mejorar el tiempo de trabajo y ahorro de agua, donde se logró con éxito el monitoreo de la humedad de suelo, dando la visualización en el aplicativo móvil, de los datos en tiempo real.

3.1.1.- Elaboracion del Diagrama en Proteus

Se elaboró un sistema informático para el control y monitoreo utilizando el programa Proteus, lo cual nos permite diseñar un modelado del prototipo Arduino, empleando los componentes que se utilizaran de manera física, en lo que se utilizaron los siguientes materiales: sensor de temperatura, sensor de humedad, sensor de lluvia, sensor de ultrasónico, el circuito Arduino uno, modulo Bluetooth, relay, diodos led, resistencias, pantalla led, una mini bomba de agua, protoboard, Por consiguiente, para comprobar el resultado de la funcionalidad del proyecto del sistema de riego, se requiere tener en cuenta las fragilidades de los sensores y de sus pines, además se elaboró la arquitectura del diagrama en Proteus para su correcto funcionamiento.

Figura 29 *Arquitectura en Proteus*

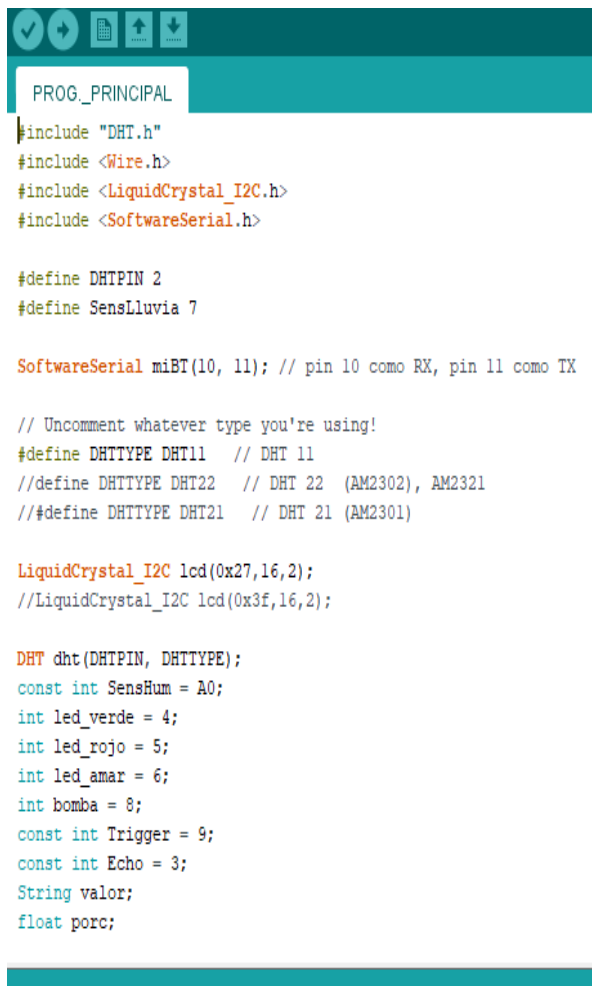


Nota: Elaboración de un prototipo informático para el control y monitoreo del sistema de riego, desarrollando la codificación en Id Arduino, donde se programa el microprocesador Arduino uno, lo cual se generará un script, para poder obtener un resultado de manera virtual y recibe valores de los diferentes sensores de temperatura, lluvia y humedad, Luego se envía a la aplicación receptora del dispositivo móvil, mediante el sensor de transferencia bluetooth – MODULO BLUETOOTH HC-06. Para luego visualizar el funcionamiento del prototipo ya con los componentes instalados, obteniendo los resultados de los sensores de lluvia, humedad y de temperatura.

3.1.1.1 Definición de librerías y variables

Se define las librerías del sensor de humedad, pantalla led y la librería de software de comunicación serial, a la vez también se define en que pines van a ir conectados los sensores, creamos las variables de humedad, las variables de los diodos led, la variable de la bomba de agua, y las variables del sensor ultrasónico.

Figura 30 *Librería y variables*



```
PROG_PRINCIPAL
#include "DHT.h"
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>

#define DHTPIN 2
#define SensLluvia 7

SoftwareSerial miBT(10, 11); // pin 10 como RX, pin 11 como TX

// Uncomment whatever type you're using!
#define DHTTYPE DHT11 // DHT 11
//define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
//#define DHTTYPE DHT21 // DHT 21 (AM2301)

LiquidCrystal_I2C lcd(0x27,16,2);
//LiquidCrystal_I2C lcd(0x3f,16,2);

DHT dht(DHTPIN, DHTTYPE);
const int SensHum = A0;
int led_verde = 4;
int led_rojo = 5;
int led_amar = 6;
int bomba = 8;
const int Trigger = 9;
const int Echo = 3;
String valor;
float porc;
```

3.1.1.2 Levantamiento de la comunicación serial

Se elabora el levantamiento de la comunicación serial. En la cual se declaran las entradas y las salidas, donde los sensores serán las salidas y los leds serán indicadores donde se visualizará un mensaje de bienvenida.

Figura 31 *Comunicación serial*

```
void setup() {  
  Serial.begin(38400);  
  miBT.begin(38400);  
  pinMode(SensLluvia, INPUT);  
  pinMode(led_verde, OUTPUT);  
  pinMode(led_rojo, OUTPUT);  
  pinMode(led_amar, OUTPUT);  
  pinMode(bomba, OUTPUT);  
  pinMode(Triquer, OUTPUT); //pin como salida  
  pinMode(Echo, INPUT); //pin como entrada  
  digitalWrite(Triquer, LOW); //Inicializamos el pin con 0  
  Serial.println(F("BIENVENIDO"));  
  dht.begin();  
  lcd.init();  
  lcd.backlight();  
  
  lcd.setCursor(2, 0);  
  lcd.print("TESIS SIACP");  
  lcd.setCursor(5, 1);  
  lcd.print("EPICI");  
  delay(4000);  
  lcd.clear();  
}
```

3.1.1.3 Funcionamiento del sensor de lluvia

Se crean variables para el sensor de lluvia y el sensor de humedad, a su vez se emplea una expresión matemática que convierte el valor analógico a porcentaje, cuando el sensor detecte que este lloviendo, el sistema se debe apagar, se apaga la bomba y se enciende el led amarillo.

Figura 32 *Funcionamiento del sensor de lluvia*

```
void loop()
{
  /*digitalWrite(bomba,LOW); //Bomba apagada
  delay(2000);
  digitalWrite(bomba,HIGH); //Bomba encendida
  delay(5000);*/
  int Lluve = digitalRead(SensLluvia);
  int HS = analogRead(SensHum);
  porc = ((1023-HS)/1023.00)*100;
  if(Lluve==0)
  {
    digitalWrite(bomba,LOW);
    valor="Esta lloviendo - Sistema de Riego Apagado";
    Serial.println(valor);
    lcd.setCursor(1, 0);
    lcd.print("Esta lloviendo");
    lcd.setCursor(0, 1);
    lcd.print("Sist. riego OFF");
    digitalWrite(led_verde, LOW);
    digitalWrite(led_amar, HIGH);
    digitalWrite(led_rojo, LOW);
  }
```

3.1.1.3 Funcionamiento del sensor de Humedad si es Mayor o igual a 900

Si el sensor de humedad de suelo es mayor o igual que 900, indicará que el suelo está seco, y se visualizará en el Lcd, y en el dispositivo móvil, la cantidad del porcentaje de humedad de suelo,

Figura 33 *Sensor de humedad*

```
else
{
  if (HS>=900)
  {
    digitalWrite(bomba,HIGH);
    lcd.setCursor(1, 0);
    lcd.print("Suelo seco  ");
    lcd.setCursor(0, 1);
    lcd.print(String(porc)+"%");
    Serial.print("Suelo seco: ");
    Serial.print(String(porc)+"%");
    valor="Suelo seco: "+String(porc)+"%";
    Serial.println();
    //delay(10);
    digitalWrite(led_verde, LOW);
    digitalWrite(led_amar, LOW);
    digitalWrite(led_rojo, HIGH);
  }
```

3.1.1.4 Funcionamiento del sensor de Humedad si es Mayor o igual a 250

Si el sensor de humedad de suelo es mayor o igual 250 y menor igual a 900, se mostrará el mensaje en el Lcd y en dispositivo móvil que el suelo se encuentra húmedo y su respectivo porcentaje.

Figura 34 *Suelo húmedo*

```
if(HS>250 && HS<900)
{
    digitalWrite(bomba,HIGH);
    lcd.setCursor(1, 0);
    lcd.print("Suelo húmedo  ");
    lcd.setCursor(0, 1);
    lcd.print(String(porc)+"%");
    valor="Suelo húmedo: "+String(porc)+"%";
    Serial.print("Suelo húmedo: ");
    Serial.print(porc);
    Serial.println(" %");
    digitalWrite(led_verde, HIGH);
    digitalWrite(led_amar, LOW);
    digitalWrite(led_rojo, HIGH);
}
```

3.1.1.5 Funcionamiento del sensor de Humedad si es Menor o igual a 250

Si el sensor de humedad de suelo es menor o igual a 250 se mostrará el mensaje en el Lcd y en el dispositivo móvil que el suelo está muy húmedo con su respectivo porcentaje

Figura 35 *Suelo muy Húmedo*

```
if(HS<=250)
{
    digitalWrite(bomba,LOW);
    lcd.setCursor(0, 0);
    lcd.print("Suelo muy húmedo");
    lcd.setCursor(0, 1);
    lcd.print(String(porc)+"%");
    Serial.print("SUELO MUY HUMEDO: ");
    Serial.print(porc);
    Serial.println(" %");
    valor="Suelo muy húmedo: "+String(porc)+"%";
    digitalWrite(led_verde, HIGH);
    digitalWrite(led_amar, LOW);
    digitalWrite(led_rojo, LOW);
}
}
```

3.1.1.4 Funcionamiento del sensor DTH

Para finalizar se lee el sensor DTH, donde se almacena el sensor de humedad, temperatura y su confirmación, si el sensor no funciona correctamente se mostrará el mensaje “no se pudo leer el sensor” y se mostrara el porcentaje del sensor DTH.

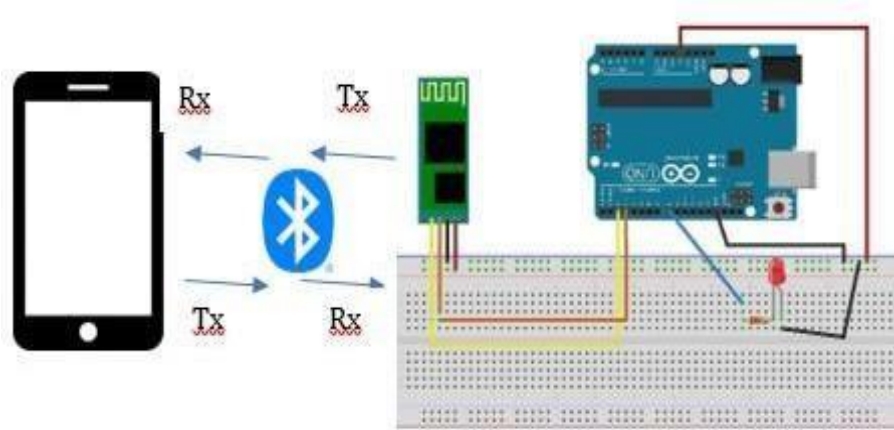
Figura 36 *Sensor dth*

```
float h = dht.readHumidity();  
float t = dht.readTemperature();  
float f = dht.readTemperature(true);  
  
if (isnan(h) || isnan(t) || isnan(f)) {  
    Serial.println(F("NO SE PUDO LEER EL SENSOR DHT"));  
    return;  
}  
  
if(Serial.available())  
    miBT.write(Serial.read());  
if(miBT.available())  
    Serial.write(miBT.read());  
valor=valor + "\nTemperatura: "+String(t)+"C";  
miBT.write("#");  
miBT.write(&valor[0]);  
miBT.write("~");  
  
delay(1000);
```

3.1.2.-Transferencia Bluetooth

La documentación existente y ejemplos investigados sobre la conexión del microcontrolador Arduino con Aplicaciones con sistema operativo Android, por bluetooth, se basan en la transferencia por los puertos seriales Tx y Rx, se conecta de manera inversa, el Arduino con el componente de Bluetooth envía mensaje con datos por el serial Tx y la aplicación recibe estos datos por su serial Rx, de igual manera es el envío de datos de la aplicación móvil hacia el microcontrolador Arduino.

Figura 37 *Transferencia bluetooth*



3.1.3.-Desarrollo de la Aplicación Móvil

Elaboración de un aplicativo móvil en Android Studio, que permitirá registrar al usuario, con su respectiva contraseña de seguridad el cual almacenará la información de lo que se muestra en el Arduino, para poder así visualizar en tiempo real los datos que nos arrojar el prototipo.

3.1.3.1 Configuración de permisos Bluetooth en Android

Para hacer el uso del módulo Bluetooth en Android Studio, se tuvo que agregar los permisos de acceso haciendo el uso de las API del Bluetooth, permitiéndonos que la aplicación móvil se conecte de manera inalámbrica con nuestro prototipo en Arduino.

Figura 38 *Conexión bluetooth*

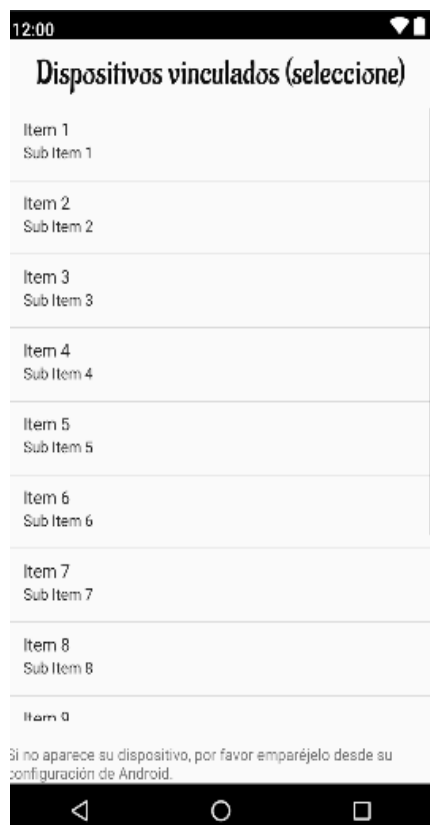
```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-feature android:name="android.hardware.bluetooth" android:required="true"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
<uses-permission android:name="android.permission.BLUETOOTH_ADVERTISE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
```

Fuente: (Android,2021)

3.1.3.2 Creación de lista de dispositivos a vincular

En esta sección se diseñó una interfaz usando List view, lo cual nos permitirá mostrar la lista de dispositivos activos, emparejados con el dispositivo móvil.

Figura 39 *List View*



3.1.3.3 Conexión Bluetooth

Programamos la conexión Bluetooth y recepción de datos proveniente del dispositivo móvil en tiempo real, para eso se obtuvo por crear hilos y con subprocesos ya determinados del mismo Android Studio, en la cual se hizo la verificación de la base de datos, se elaboró el llenado de listas e inicialización de los arreglos, se obtuvo la dirección Mac del dispositivo y se pide al usuario que encienda el Bluetooth si está apagado.

Figura 40 Verificación de la BD

```
public class actividad_lista_dispositivos extends AppCompatActivity {

    TextView txtestadoconexion;
    public static String EXTRA_DEVICE_ADDRESS = "device_address";
    private BluetoothAdapter btadaptadon;
    private ArrayAdapter<String> dispositivosarray;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_lista_dispositivos);
    }

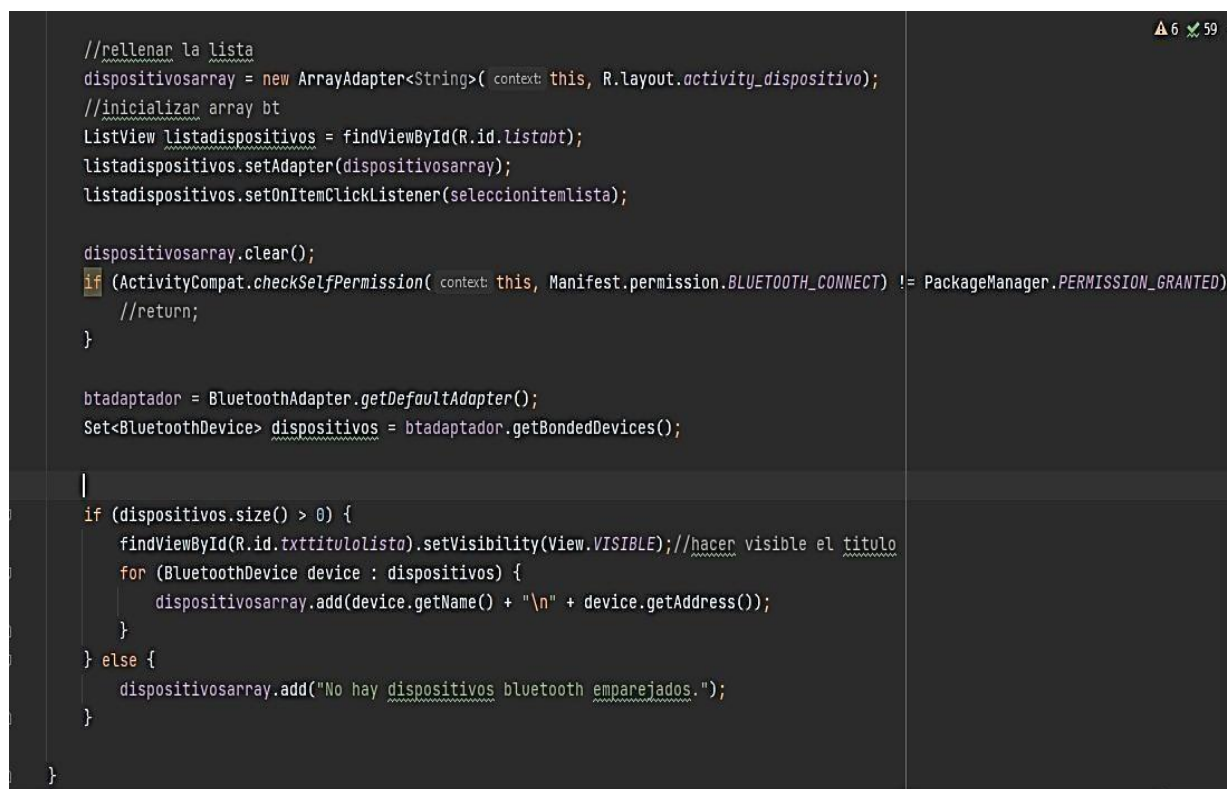
    @Override
    public void onResume() {
        super.onResume();
        chekarestadoBT();

        txtestadoconexion = findViewById(R.id.txtconexion);
        txtestadoconexion.setTextSize(30);
        txtestadoconexion.setText(" ");
    }
}
```

3.1.3.4 Llenado de arreglos para la lista de dispositivos.

Se elabora el llenado en la lista y se inicializa el arreglo, después se utiliza un clear, para borrar la matriz y que los elementos no se dupliquen al reanudar desde el onPause, para después obtener el adaptador Bluetooth local con un conjunto de dispositivos emparejados y agregados a la lista.

Figura 41 Llenado de listas



```
//rellenar la lista
dispositivosarray = new ArrayAdapter<String>( context: this, R.layout.activity_dispositivo);
//inicializar array bt
ListView listadispositivos = findViewById(R.id.listabt);
listadispositivos.setAdapter(dispositivosarray);
listadispositivos.setOnItemClickListener(seleccionitemlista);

dispositivosarray.clear();
if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED)
    //return;
}

btadaptador = BluetoothAdapter.getDefaultAdapter();
Set<BluetoothDevice> dispositivos = btadaptador.getBondedDevices();

if (dispositivos.size() > 0) {
    findViewById(R.id.txttitulolista).setVisibility(View.VISIBLE); //hacer visible el titulo
    for (BluetoothDevice device : dispositivos) {
        dispositivosarray.add(device.getName() + "\n" + device.getAddress());
    }
} else {
    dispositivosarray.add("No hay dispositivos bluetooth emparejados.");
}
}
```

3.1.3.5 Dirección MAC

Se obtiene la dirección MAC del dispositivo, que son los últimos 17 caracteres en la vista.

Figura 42 Dirección Mac

```
private OnItemClickListener seleccionitemlista = new OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> av, View v, int i1, long i2) {
        txttestadoconexion.setText("Conectando...");

        String info = ((TextView) v).getText().toString();
        String direccion = info.substring(info.length() - 17);
        |
        Intent i = new Intent( packageContext: activity_lista_dispositivos.this, Control.class);
        i.putExtra(EXTRA_DEVICE_ADDRESS, direccion);
        startActivity(i);
    }
};
```

3.1.3.6 Permiso Usuario

Este método nos permite poder verificar si el dispositivo tiene Bluetooth y si esta encendido, caso contrario se le pide al usuario que lo encienda.

Figura 43 Permiso al usuario

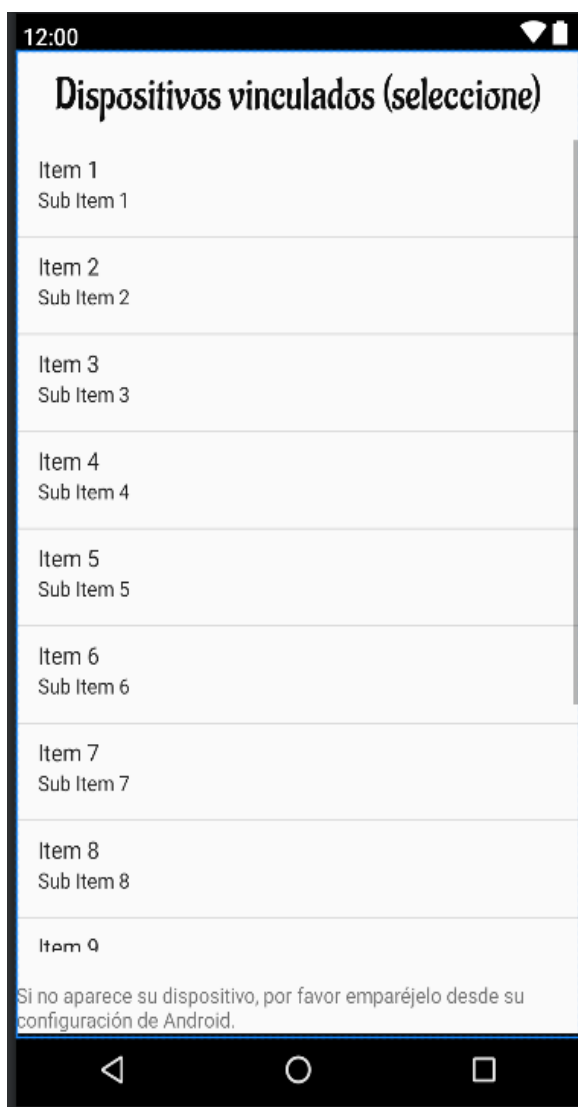
```
private void chekarestadoBT() {

    btadaptador = BluetoothAdapter.getDefaultAdapter();
    if (btadaptador == null) {
        Toast.makeText(getBaseContext(), text: "El dispositivo no soporta Bluetooth", Toast.LENGTH_SHORT).show();
        finish();
    } else {
        if (!btadaptador.isEnabled()) {
            |
            Intent habilitaBT = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
            if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED)
                return;
            }
            startActivityForResult(habilitaBT, requestCode: 1);
        }
    }
}
```

3.1.3.7 Dispositivos a vincular

De esta manera quedará la selección de dispositivos a vincular los cuales se mostrarán en nuestra lista, cada ítem representa el nombre de un dispositivo diferente.

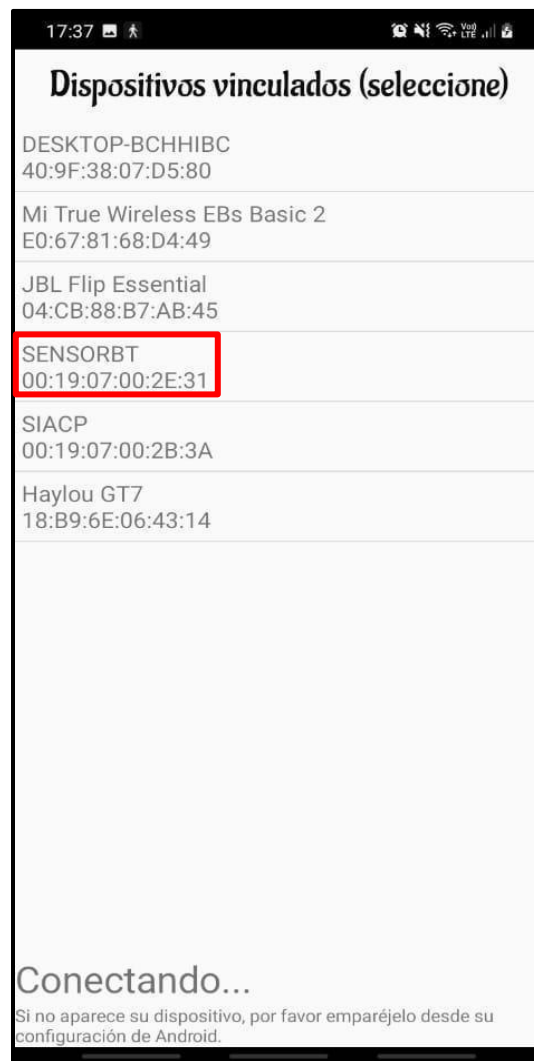
Figura 44 *Dispositivos a vincular*



3.1.3.8 Conexión Bluetooth SENSORBT

Una vez iniciado la aplicación podremos visualizar una interfaz con la lista de los dispositivos disponibles, en este caso buscamos en la lista SENSORBT, el cual es el nombre de nuestro módulo Bluetooth del Arduino. En caso contrario el dispositivo no puede vincularse, en la parte inferior nos mostrará un mensaje “si no aparece su dispositivo. Por favor emparéjelo desde su configuración en Android”.

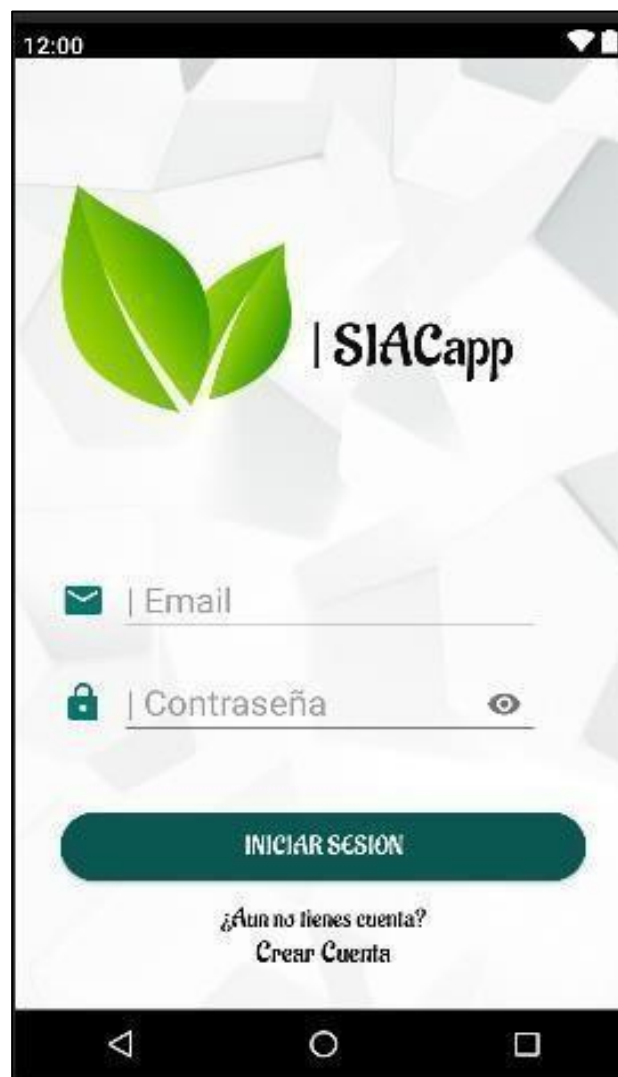
Figura 45 *Dispositivo SESORRBT*



3.1.3.9 Inicio de Sesión

Una vez vinculado nuestra aplicación con el módulo bluetooth del Arduino SENSORT, nos mostrara una interfaz de inicio de sesión, la cual tiene como finalidad cumplir con los requerimientos funcionales planteado al inicio del proyecto, la cual consiste en la seguridad y privacidad del usuario.

Figura 46 *Pantalla inicio*



3.1.3.10 Registro de usuario

En esta interfaz, como el usuario no dispone con una cuenta, se procede a realizar el registro, solicitando sus datos y hacer el llenado en los siguientes campos vacíos, y registrándose en la base de datos.

Figura 47 Interfaz registro



12:00

 **SIACapp**

Nombre Completo

Usuario

Correo electrónico

Contraseña 

REGISTRAR

¿Ya tienes cuenta? [Iniciar](#)

Figura 48 Registro usuario



11:42

 **SIACapp**

Nombre Completo
juan

Usuario
carlos

Correo electrónico
juan123@gmail.com

Contraseña
**** 

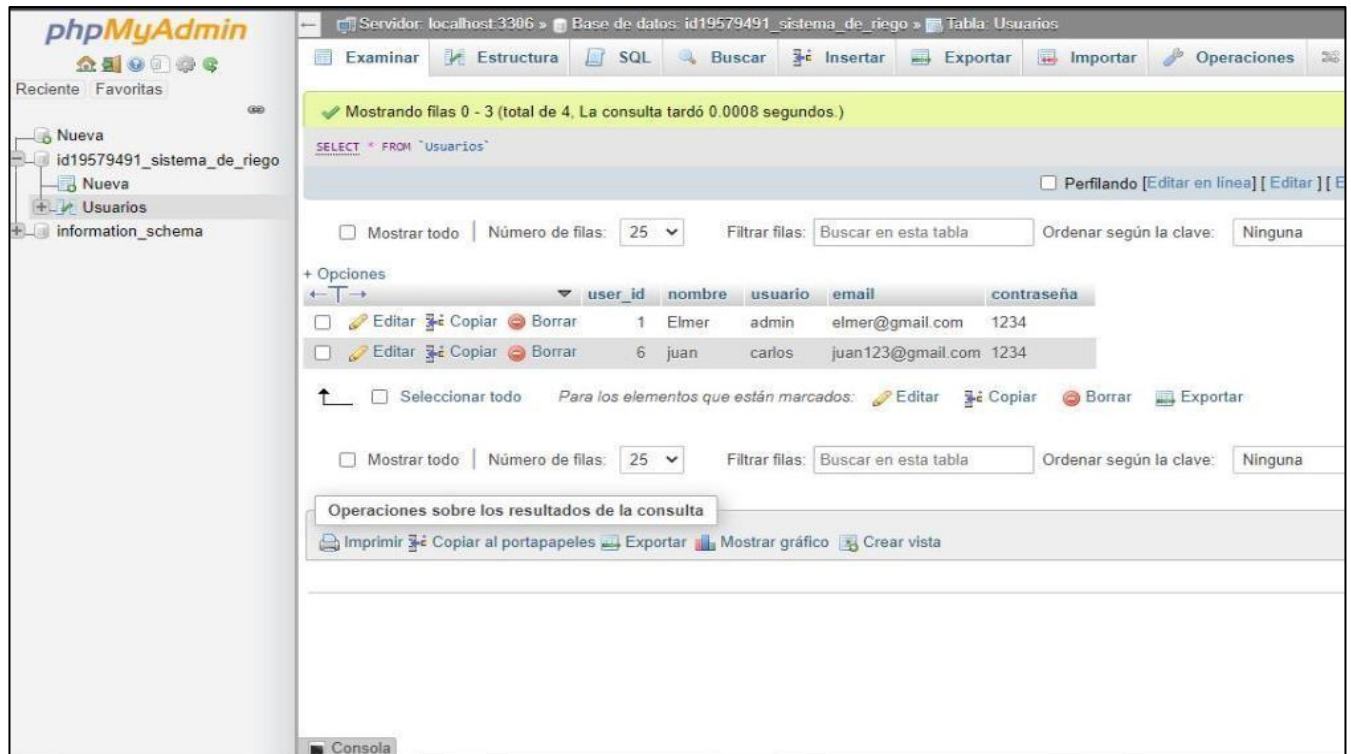
REGISTRAR

¿Ya tienes cuenta? [Iniciar](#)

3.1.3.11 Registro en la Base de Datos

Una vez ingresado los datos del usuario al aplicativo móvil, podemos ver que automáticamente los datos quedan registrados en nuestra base de datos, lo cual le permitirá poder iniciar sesión en nuestra app.

Figura 49 Registro de datos



3.1.3.12 Inicio de Sesión

En esta interfaz de logeo o inicio de sesión, nos pedirá nuestro email y contraseña, que con anterioridad hemos registrado y almacenado en nuestra base de datos. Llenamos los campos vacíos y pulsamos en el botón de INICIAR SESION.

Figura 50 *Logeo*



3.1.3.13 Menú de opciones

Luego de haber iniciado sesión, la siguiente interfaz es de nuestro menú de opciones el cual podremos visualizar dos botones, el botón de control que nos permite visualizar en tiempo real los valores de los diferentes sensores, y el segundo botón de registro nos permite visualizar un pequeño texto sobre el aplicativo y los desarrolladores.

Figura 51 *Menú opciones*



3.1.3.14 Botón registro

Esta interfaz nos presenta la información que encontraremos al pulsar el botón de registro/ bienvenida, donde se visualiza una breve información sobre el aplicativo móvil, y los desarrolladores de la aplicación.

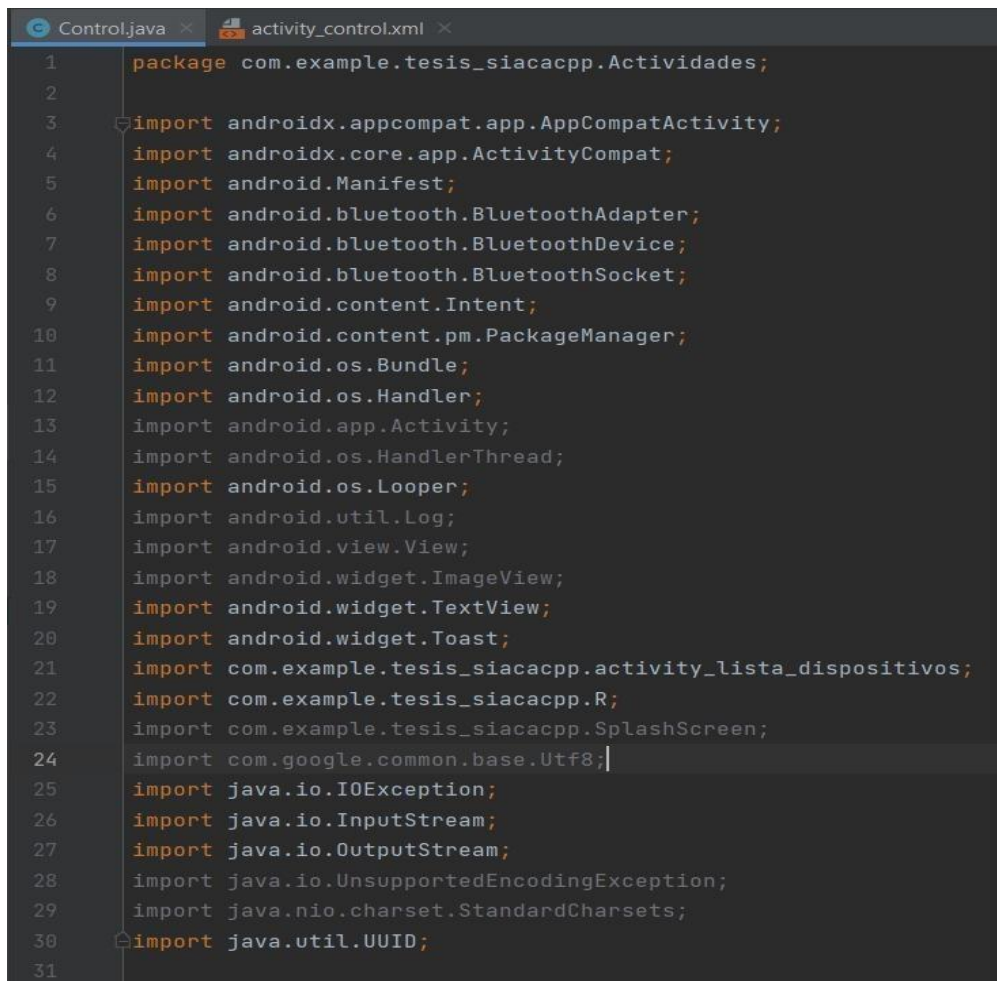
Figura 52 Botón registro/ bienvenida



3.1.3.15 Codificación del botón control

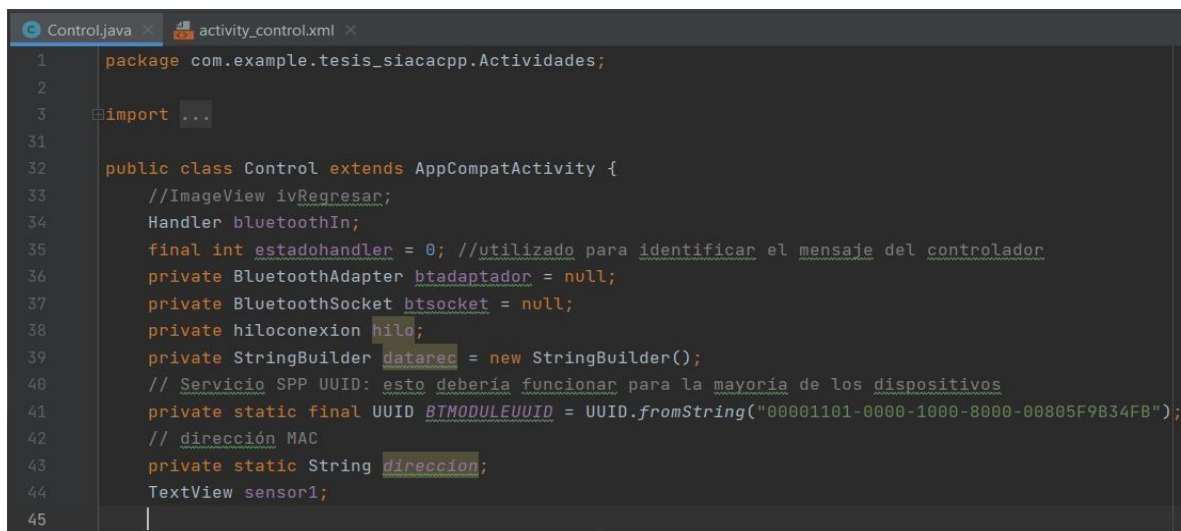
Para recepcionar y mostrar la información en tiempo real que envía el dispositivo Arduino al aplicativo móvil, y establecer una comunicación entre ambos. Se utilizó el servicio UUID (Universal Unique Identifier), el cual nos permite identificar un servicio que nos brinda el dispositivo Bluetooth, ya sea en estándar que nos permite conectarnos al módulo Bluetooth haciendo el uso de BASE_UUID: 00000000-0000-1000-8000-00805F9B34FB. Así como se mostrará en las siguientes imágenes.

Figura 53 Librerías Botón Control



```
1 package com.example.tesis_siacacpp.Actividades;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import androidx.core.app.ActivityCompat;
5 import android.Manifest;
6 import android.bluetooth.BluetoothAdapter;
7 import android.bluetooth.BluetoothDevice;
8 import android.bluetooth.BluetoothSocket;
9 import android.content.Intent;
10 import android.content.pm.PackageManager;
11 import android.os.Bundle;
12 import android.os.Handler;
13 import android.app.Activity;
14 import android.os.HandlerThread;
15 import android.os.Looper;
16 import android.util.Log;
17 import android.view.View;
18 import android.widget.ImageView;
19 import android.widget.TextView;
20 import android.widget.Toast;
21 import com.example.tesis_siacacpp.activity_lista_dispositivos;
22 import com.example.tesis_siacacpp.R;
23 import com.example.tesis_siacacpp.SplashScreen;
24 import com.google.common.base.UTF8;
25 import java.io.IOException;
26 import java.io.InputStream;
27 import java.io.OutputStream;
28 import java.io.UnsupportedEncodingException;
29 import java.nio.charset.StandardCharsets;
30 import java.util.UUID;
```

Figura 54 Declarando variables y UUID



```
1 package com.example.tesis_siacacpp.Actividades;
2
3 import ...
31
32 public class Control extends AppCompatActivity {
33     //ImageView ivRegresar;
34     Handler bluetoothIn;
35     final int estadohandler = 0; //utilizado para identificar el mensaje del controlador
36     private BluetoothAdapter btadaptador = null;
37     private BluetoothSocket btsocket = null;
38     private hiloconexion hilo;
39     private StringBuilder datarec = new StringBuilder();
40     // Servicio SPP UUID: esto debería funcionar para la mayoría de los dispositivos
41     private static final UUID BTMODULEUUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
42     // dirección MAC
43     private static String direccion;
44     TextView sensor1;
45 }
```

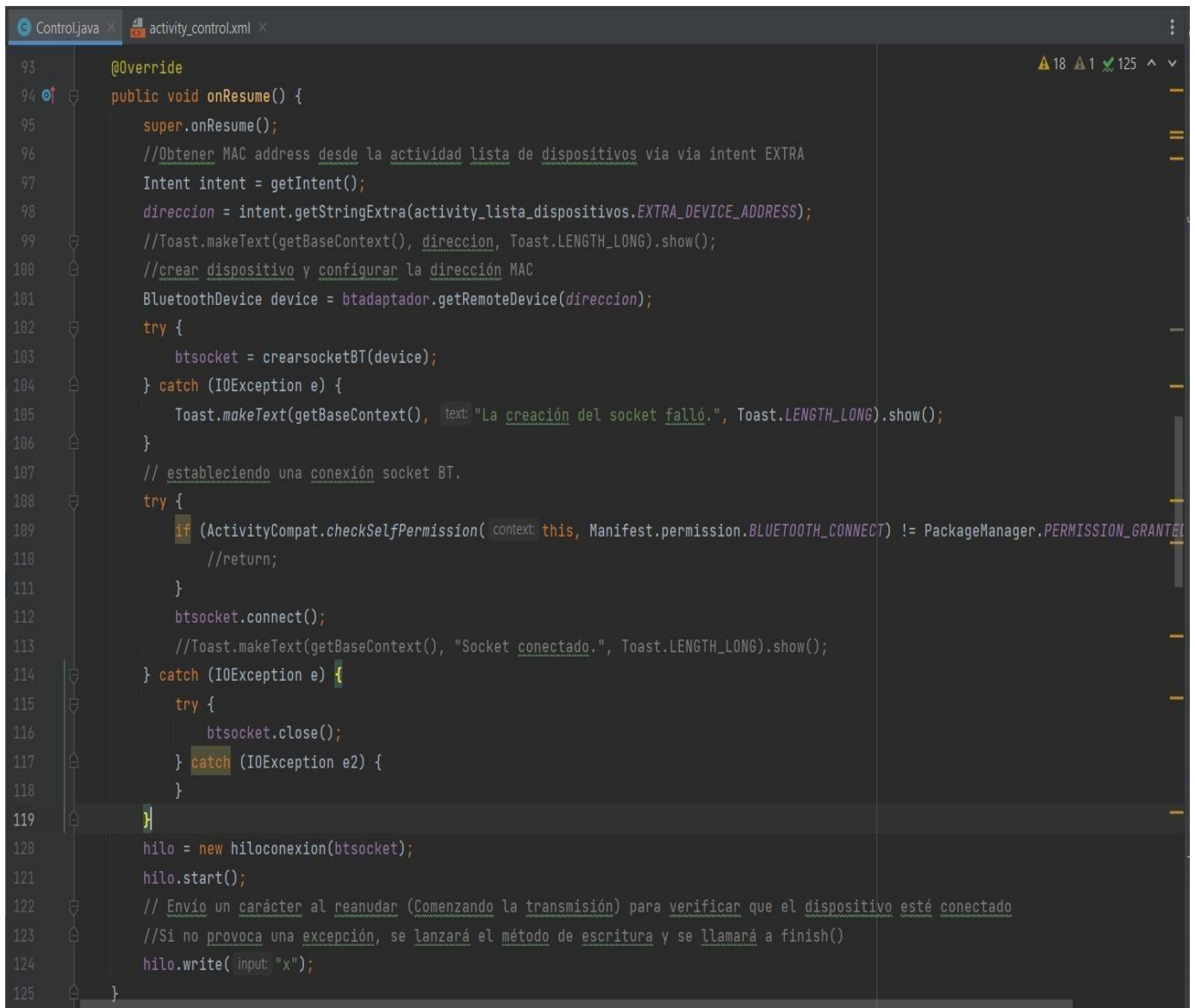
Figura 55 Método Handle

```
Control.java x activity_control.xml x
43 private static String direccion;
44 TextView sensor1;
45
46 @Override
47 protected void onCreate(Bundle savedInstanceState) {
48     super.onCreate(savedInstanceState);
49     setContentView(R.layout.activity_control);
50     //para un boton atras en el action bar
51     //getSupportActionBar().setDisplayHomeAsUpEnabled(true);
52     sensor1=(TextView)findViewById(R.id.txtsensor1);
53     bluetoothIn = new Handler(Looper.getMainLooper())
54     {
55         @Override
56         public void handleMessage(android.os.Message msg) {
57             if (msg.what == estadohandler) { //si el mensaje es lo que queremos
58                 String rmensaje = (String) msg.obj; //msg.arg1 = bytes del subproceso de conexión
59                 datarec.append(rmensaje); // sigue agregando a la cadena hasta que llegue a ~
60                 int finlinea = datarec.indexOf("~"); // determinar el final de la línea
61                 if (finlinea > 0) { //verificar de que haya datos antes de ~
62                     if (datarec.charAt(0) == '#') { //si empieza por # sabemos que es lo que buscamos
63                         //extraer datos individuales con separador, mostrar datos
64                         //String [] ar = dataInPrint.split("[+]");
65                         sensor1.setText(datarec.substring(1,datarec.length()-1));
66                     }
67                     datarec.delete(0, datarec.length()); //borrar la data
68                 }
69             }
70         }
71     };
72     btadaptador = BluetoothAdapter.getDefaultAdapter(); //adaptador BT por defecto
73     chekBTState();
74 }
```

Figura 56 Conexión bluetooth usando UUID

```
Control.java x activity_control.xml x
82 }
83
84
85 private BluetoothSocket crearsocketBT(BluetoothDevice device) throws IOException {
86     // crea una conexión saliente segura con el dispositivo BT usando UUID
87     if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED)
88         //return null;
89     }
90     return device.createRfcommSocketToServiceRecord(BTMODULEUUID);
91 }
92 }
```

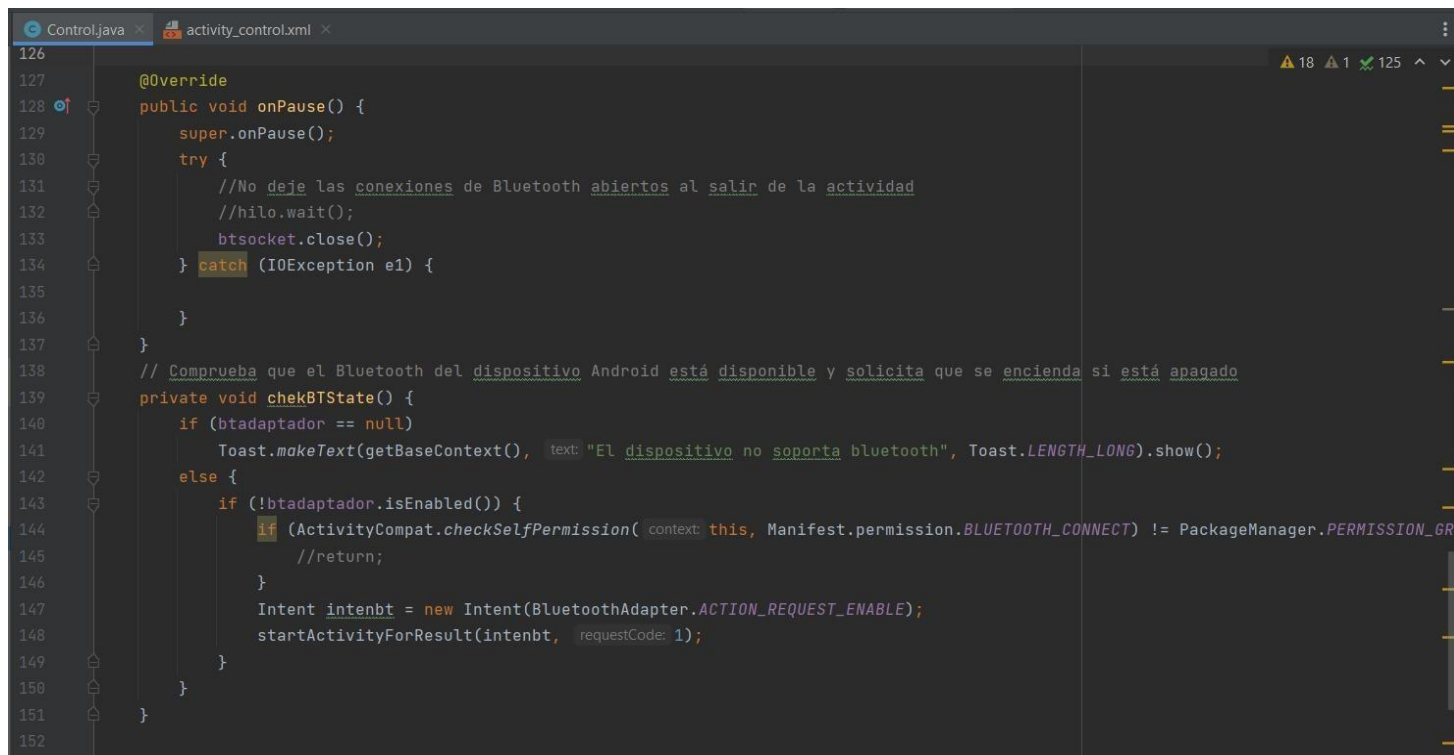

Figura 57 *Obtener listado de dispositivos Bluetooth*



The image shows a screenshot of an IDE with two tabs: 'Control.java' and 'activity_control.xml'. The 'Control.java' tab is active, displaying Java code for an Android application. The code is for the `onResume()` method of a class. It includes comments in Spanish and code for obtaining a Bluetooth MAC address from an intent, creating a `BluetoothDevice` object, and attempting to connect to it via a `btsocket`. The code uses `Toast` for user feedback and includes a permission check for `Manifest.permission.BLUETOOTH_CONNECT`. The code is as follows:

```
93  @Override
94  public void onResume() {
95      super.onResume();
96      //Obtener MAC address desde la actividad lista de dispositivos via intent EXTRA
97      Intent intent = getIntent();
98      direccion = intent.getStringExtra(activity_lista_dispositivos.EXTRA_DEVICE_ADDRESS);
99      //Toast.makeText(getBaseContext(), direccion, Toast.LENGTH_LONG).show();
100     //crear dispositivo y configurar la dirección MAC
101     BluetoothDevice device = btadaptador.getRemoteDevice(direccion);
102     try {
103         btsocket = crearsocketBT(device);
104     } catch (IOException e) {
105         Toast.makeText(getBaseContext(), text: "La creación del socket falló.", Toast.LENGTH_LONG).show();
106     }
107     // estableciendo una conexión socket BT.
108     try {
109         if (ActivityCompat.checkSelfPermission(context: this, Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED)
110             //return;
111         }
112         btsocket.connect();
113         //Toast.makeText(getBaseContext(), "Socket conectado.", Toast.LENGTH_LONG).show();
114     } catch (IOException e) {
115         try {
116             btsocket.close();
117         } catch (IOException e2) {
118         }
119     }
120     hilo = new hiloconexion(btsocket);
121     hilo.start();
122     // Envío un carácter al reanudar (Comenzando la transmisión) para verificar que el dispositivo esté conectado
123     //Si no provoca una excepción, se lanzará el método de escritura y se llamará a finish()
124     hilo.write(input: "x");
125 }
```

Figura 58 Creación de eventos

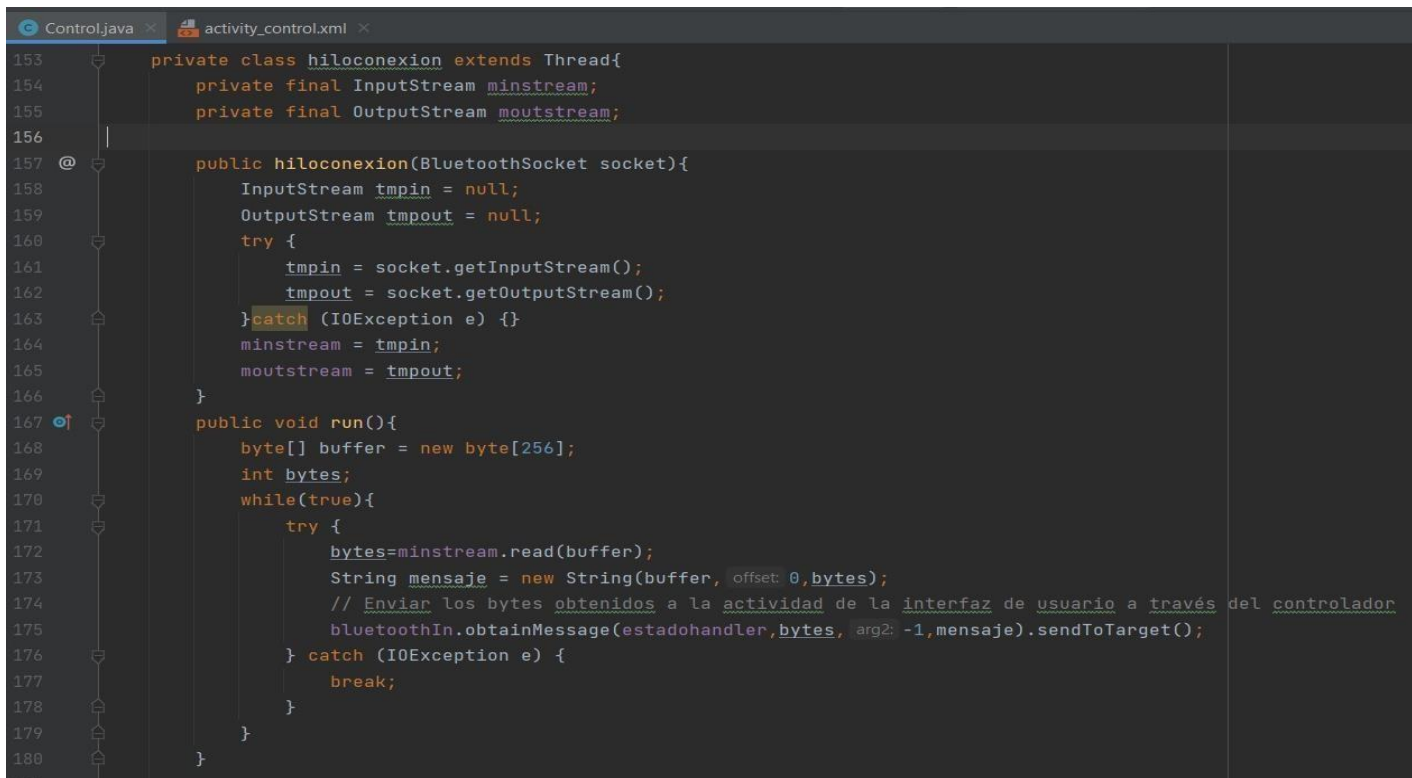


```

126
127 @Override
128 public void onPause() {
129     super.onPause();
130     try {
131         //No deje las conexiones de Bluetooth abiertos al salir de la actividad
132         //hilo.wait();
133         btsocket.close();
134     } catch (IOException e1) {
135
136     }
137 }
138 // Comprueba que el Bluetooth del dispositivo Android está disponible y solicita que se encienda si está apagado
139 private void checkBTState() {
140     if (btadaptador == null)
141         Toast.makeText(getBaseContext(), "El dispositivo no soporta bluetooth", Toast.LENGTH_LONG).show();
142     else {
143         if (!btadaptador.isEnabled()) {
144             if (ActivityCompat.checkSelfPermission(context: this, Manifest.permission.BLUETOOTH_CONNECT) != PackageManager.PERMISSION_GRANTED) {
145                 //return;
146             }
147             Intent intenbt = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
148             startActivityForResult(intenbt, requestCode: 1);
149         }
150     }
151 }
152

```

Figura 59 Creación de hilos



```

153 private class hiloconexion extends Thread{
154     private final InputStream minstream;
155     private final OutputStream moutstream;
156
157     @
158     public hiloconexion(BluetoothSocket socket){
159         InputStream tmpin = null;
160         OutputStream tmpout = null;
161         try {
162             tmpin = socket.getInputStream();
163             tmpout = socket.getOutputStream();
164         } catch (IOException e) {}
165         minstream = tmpin;
166         moutstream = tmpout;
167     }
168     public void run(){
169         byte[] buffer = new byte[256];
170         int bytes;
171         while(true){
172             try {
173                 bytes=minstream.read(buffer);
174                 String mensaje = new String(buffer, offset: 0,bytes);
175                 // Enviar los bytes obtenidos a la actividad de la interfaz de usuario a través del controlador
176                 bluetoothIn.obtainMessage(estadohandler,bytes, arg2: -1,mensaje).sendToTarget();
177             } catch (IOException e) {
178                 break;
179             }
180         }
181     }
182 }
183

```


Figura 60 Mensaje Conexión

```

182 @
183     public void write(String input){
184         //StandardCharsets.UTF_8
185         try {
186             moutstream.write(input.getBytes());
187         } catch (IOException e) {
188             //si no se puede escribir, cerrar la aplicación
189             Toast.makeText(getBaseContext(), text: "Conexión fallida", Toast.LENGTH_LONG).show();
190             finish();
191         }
192     }
193 }

```

Figura 61 Código XML del botón control

The image shows the Android Studio interface with the XML code for 'activity_control.xml' on the left and a preview of the UI on the right.

XML Code:

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res-
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:background="#12FFFFFF"
8   tools:context=".Actividades.Control">
9
10  <ImageView
11
12      android:layout_width="match_parent"
13      android:layout_height="match_parent"
14      android:contentDescription="Usuario"
15      android:scaleType="centerCrop"
16      android:src="@drawable/gray" />
17
18  <LinearLayout
19      android:layout_width="match_parent"
20      android:layout_height="match_parent"
21      android:background="#12FFFFFF"
22      android:gravity="center"
23      android:orientation="vertical"
24      android:paddingRight="30dp">
25
26
27      <TextView
28          android:layout_width="wrap_content"
29          android:layout_height="wrap_content"
30          android:layout_marginLeft="30dp"
31          android:layout_marginRight="30dp"

```

UI Preview: The preview shows a dark screen with the text "Control Arduino" at the top. Below it, the text "Riego por Goteo" is displayed in a large, white, stylized font. At the bottom, the text "valor" is displayed in a smaller, white font.

3.1.3.16 Botón control

El siguiente botón del menú es de Control, esta interfaz nos permitirá visualizar el resultado de los valores que emite el prototipo de Arduino en tiempo real, ya sea el sensor de humedad del suelo, sensor de temperatura y el sensor de lluvia.

Figura 62 *Control Arduino*



3.1.3.17 Botón control – Sensor humedad (suelo seco)

En esta interfaz, el sensor de humedad del suelo, puesto en tierra seca, mostrara en tiempo real un porcentaje de 5.38% de humedad y un 23.30 °C de temperatura de ambiente.

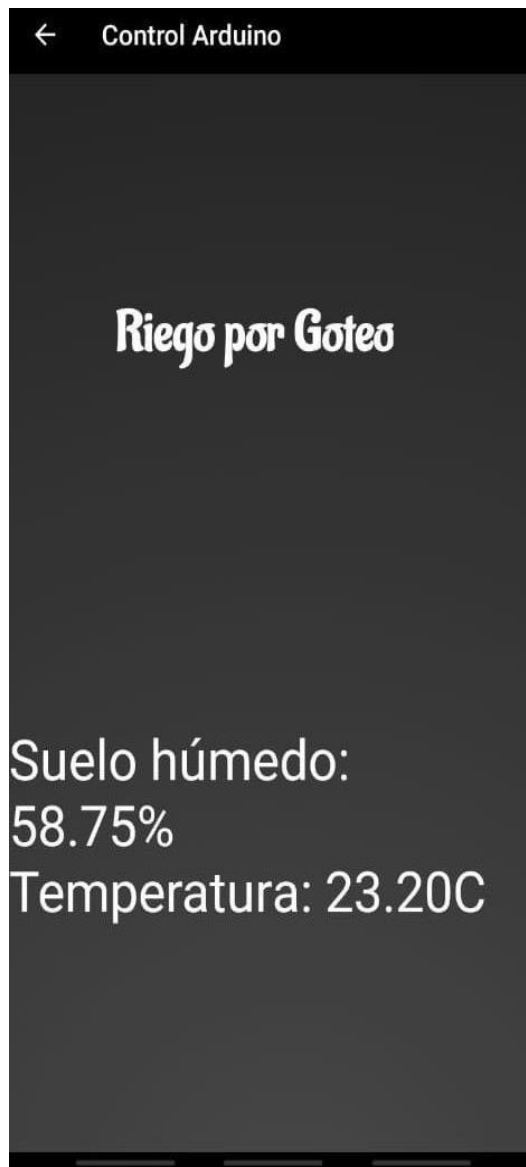
Figura 63 *Resultado suelo seco*



3.1.3.18 Botón control – Sensor humedad (suelo húmedo)

En esta ilustración se presentan los resultados del sensor humedad del suelo, en la que el suelo se le hace un agregado moderado de agua, esperamos que la tierra humedezca y así nos arroja un porcentaje de 58.75% de humedad y la temperatura ambiente se mantiene en un 23.20 °C.

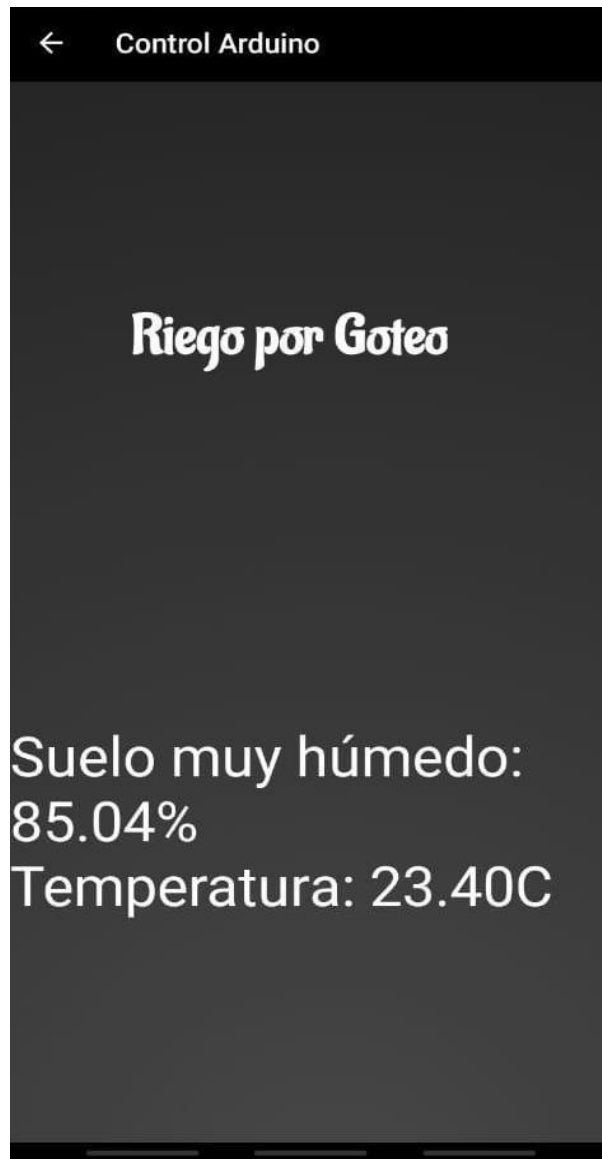
Figura 64 *Resultado suelo húmedo*



3.1.3.19 Botón Control – Sensor humedad (muy húmedo)

Para finalizar con las pruebas se agrega más agua, con el fin de cubrir la superficie de la tierra, con esto los resultados del sensor humedad del suelo nos mostrara un porcentaje de 85.04% de humedad y así mismo la temperatura de ambiente se mantiene en un 23.40 °C.

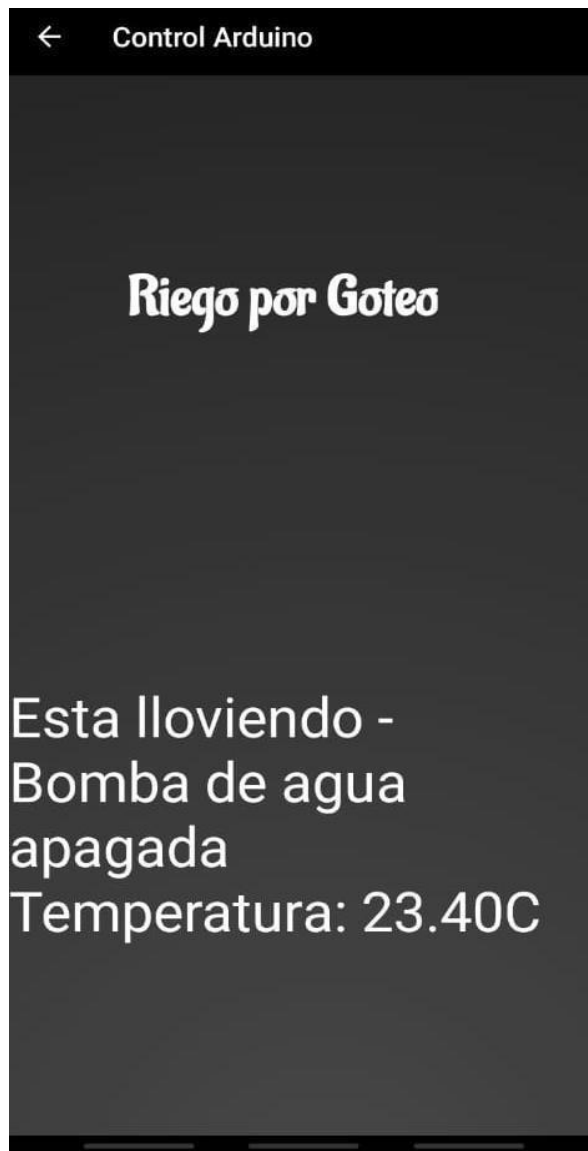
Figura 65 *Resultado suelo muy húmedo*



3.1.3.20 Botón Control - Sensor Lluvia

Este sensor tiene la funcionalidad de detectar un riego externo que no sea generado, en este caso sea la lluvia. De tal forma que la bomba de agua se apaga automáticamente y nos aparece un mensaje en la aplicación “está lloviendo – bomba de agua apagada” con su respectiva temperatura de ambiente.

Figura 66 *Resultado sistema apago*

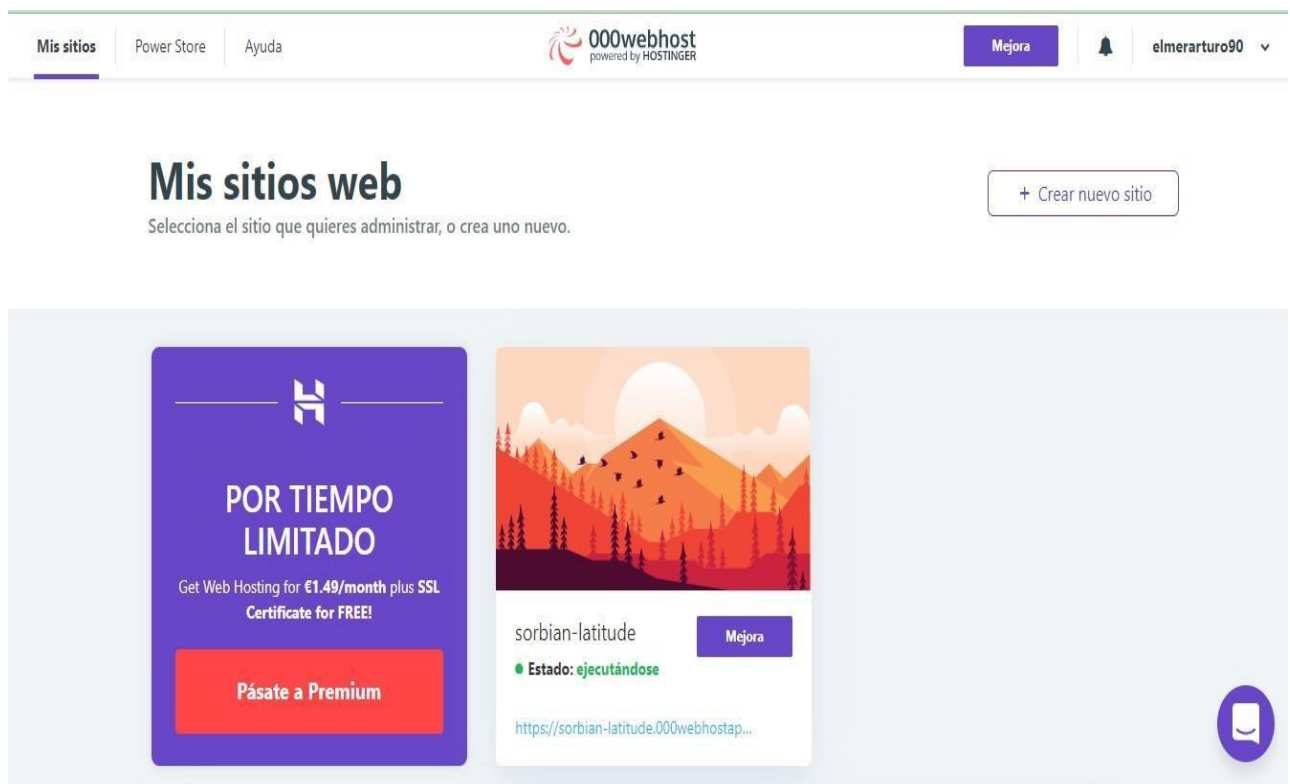


3.1.4.-Desarrollo Base de Datos

3.1. 4.1-Implementacion del hosting

Se implementó un Hosting libre en 000webhost, para la creación del nombre de dominio del aplicativo móvil, que a su vez funcionara como servidor, por lo que ya no se utilizara la dirección IP, de una red local.

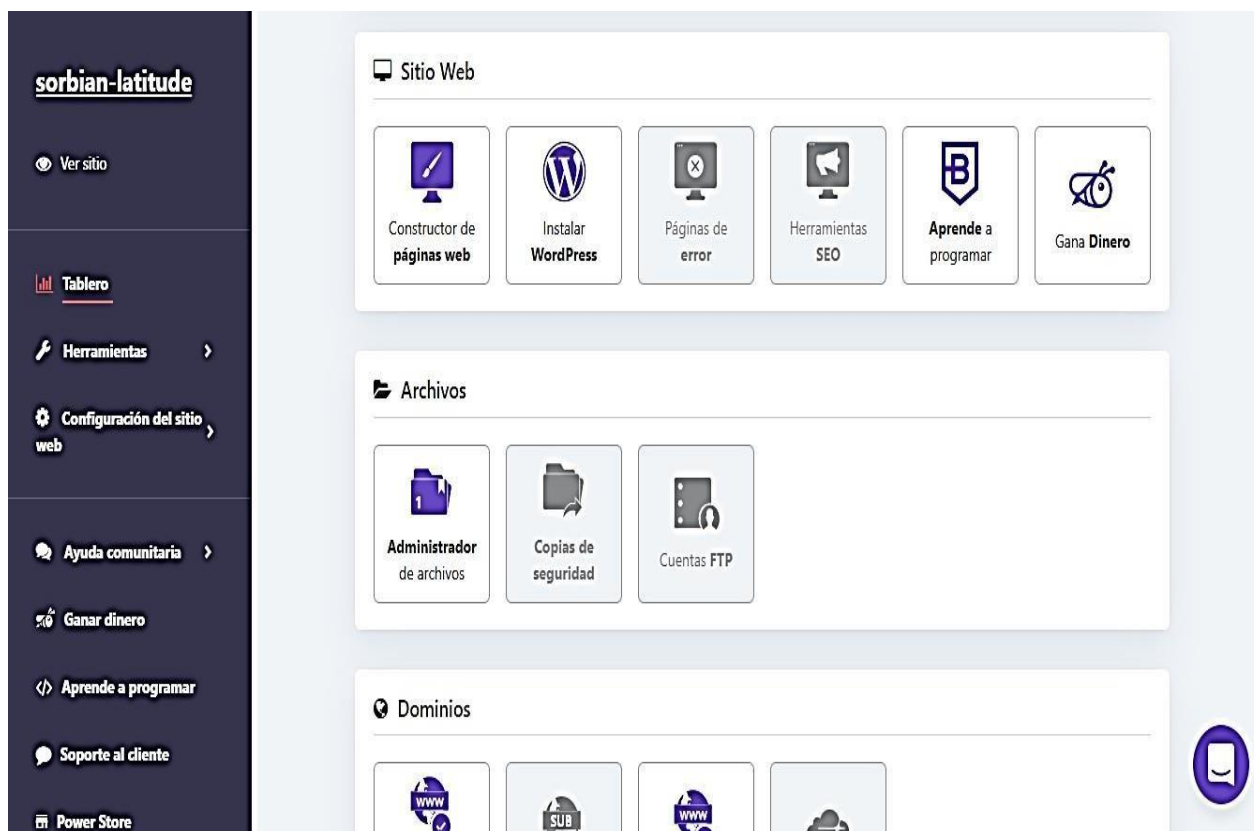
Figura 67 Hosting libre 000webhost



3.1.4.2 Administrador de archivos

En esta sección vamos a seleccionar Administrador de Archivos, para poder subir los Archivos ala base de datos.

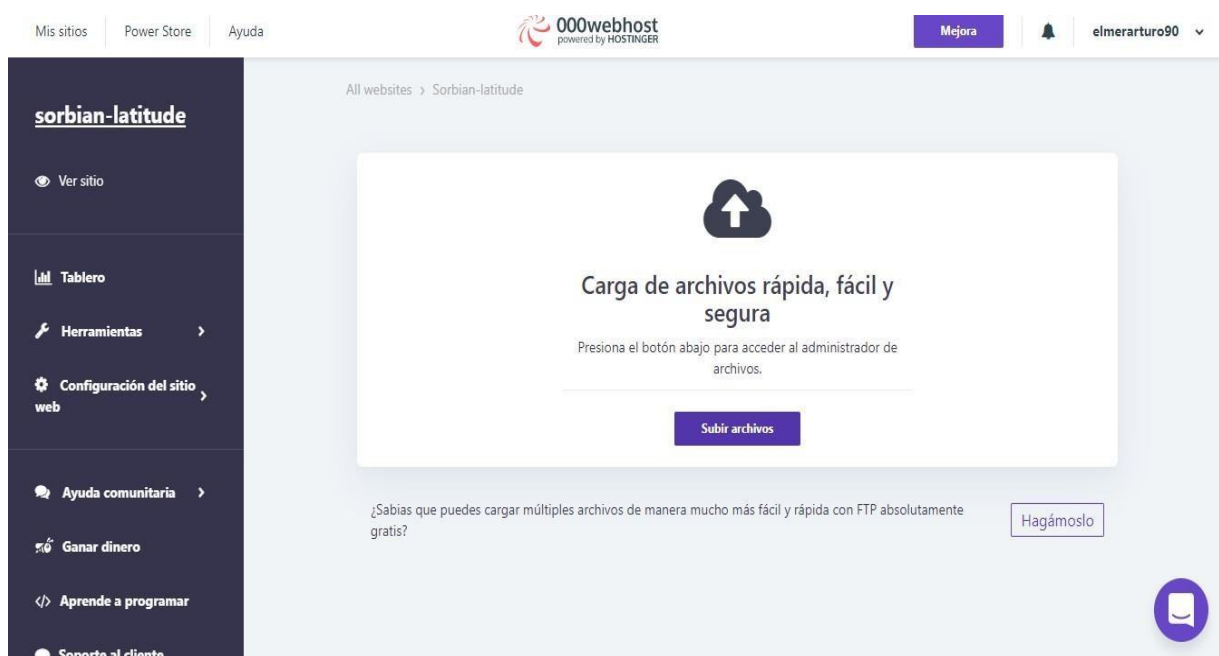
Figura 68 *Administrador de Archivos*



3.1.4.3 Importamos nuestra Base de Datos

En la siguiente ventana importamos los archivos ya creados a la nube, para la conexión de la base de datos desarrollado en el lenguaje php.

Figura 69 *Subir Archivos BD*



3.1.4.4 Código para conexión de la base de datos

Codificación de la conexión de la base de datos en php, donde se mostrará el nombre de la base de datos. el nombre del usuario de la base de datos y el nombre del host de la base de datos, para el correcto funcionamiento de la conexión.

Figura 70 *Conexión. Php*



```
1 <?php
2
3 // $conn = mysqli_connect("localhost", "root", "", "u$enlace = mysqli_connect("127.0.0.1", "mi_usuario",
4 "mi_contraseña", "mi_bd");
5
6
7 $conn = mysqli_connect("localhost", "id19579491_elmer", "Lopezacosta@1991", "id19579491_sistema_de_riego");
8
9 if (!$conn) {
10     echo "Error: No se pudo conectar a MySQL." . PHP_EOL;
11     echo "error de depuración: " . mysqli_connect_errno() . PHP_EOL;
12     echo "error de depuración: " . mysqli_connect_error() . PHP_EOL;
13     exit;
14 }
15
16 //echo "Éxito: Se realizó una conexión apropiada a MySQL! La base de datos mi_bd es genial." . PHP_EOL;
17 //echo "Información del host: " . mysqli_get_host_info($conn) . PHP_EOL;
18
19 //mysqli_close($conn);
20
21 ?>
```

3.1.4.5 Código Logeo (inicio sesión)

En esta interfaz nos muestra la codificación para el logeo/inicio sesión en la base de datos, primero se hace el llamado a la conexión de la base, la cual anteriormente lo guardamos como connexion.php, luego se desarrolla unas validaciones para que el usuario solo pueda ingresar con su email y su contraseña.

Figura 71 *Logeo.php*



```
1 <?php
2
3 require_once 'conexion.php';
4 global $conn;
5
6 $email=$_POST['email'];
7 $contrasena=$_POST['contrasena'];
8
9 //$usu_usuario="aroncal@gmail.com";
10 //$usu_password="12345678";
11
12 $sentencia=$conn->prepare("SELECT * FROM Usuarios WHERE email=? AND contraseña=?");
13 $sentencia->bind_param('ss',$email,$contrasena);
14 $sentencia->execute();
15
16 $resultado = $sentencia->get_result();
17 if ($fila = $resultado->fetch_assoc()) {
18     echo json_encode($fila,JSON_UNESCAPED_UNICODE);
19 }
20 $sentencia->close();
21 $conn->close();
22 ?>
```

3.1.4.6 Conexión de la base de datos

La siguiente ventana presenta la codificación del registro de usuario de la base de datos, de igual forma hacemos el llamado a la conexión.php. Se procede a declarar las variables (nombre, usuario, email, contraseña), la cual nos permitirá en la aplicación ingresar los datos correspondientes del usuario.

Figura 72 *Register.php*



```
1 <?php
2
3 require_once 'conexion.php';
4 global $conn;
5
6 $nombre = $_POST['nombre'];
7 $usuario = $_POST['usuario'];
8 $email = $_POST['email'];
9 $contrasena = $_POST['contrasena'];
10 //$usu_usuario="aroncal@gmail.com";
11 //$usu_password="12345678";
12
13 $consulta = "INSERT INTO Usuarios (nombre, usuario, email, contraseña) VALUES ('$nombre', '$usuario', '$email', '$contrasena')";
14 mysqli_query($conn, $consulta) or die (mysqli_error());
15 mysqli_close($conn);
16
17
18 ?>
```

3.1.4.7 Base de datos creada

Por último, entramos al Administrador de la base de datos, y obtendremos el nombre, el usuario, el host de la base de datos, quedando lista para su gestión. Por último, accedemos en PhpMyAdmin para un mejor control de la base.

Figura 73 Base de datos

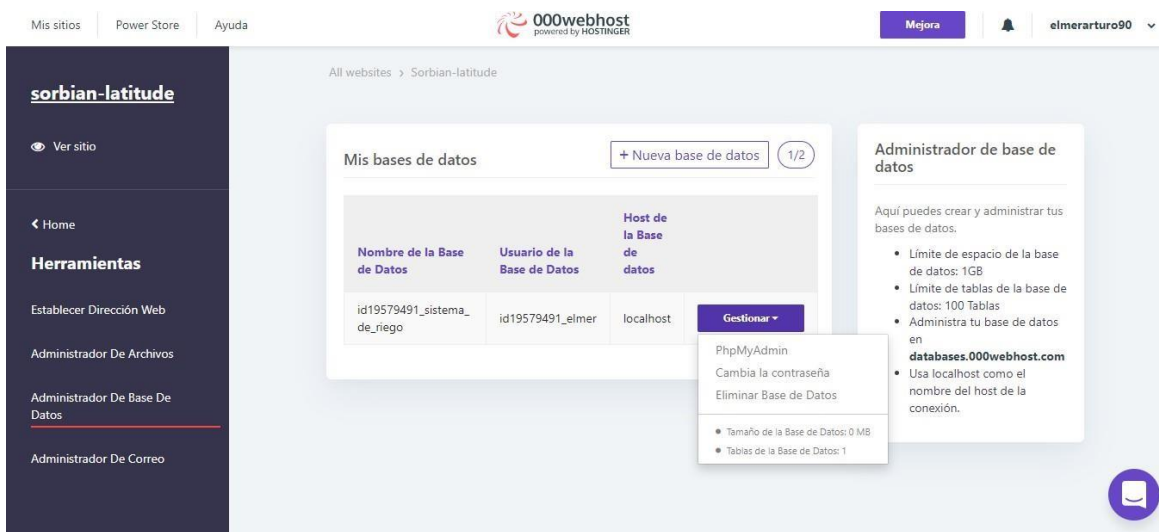
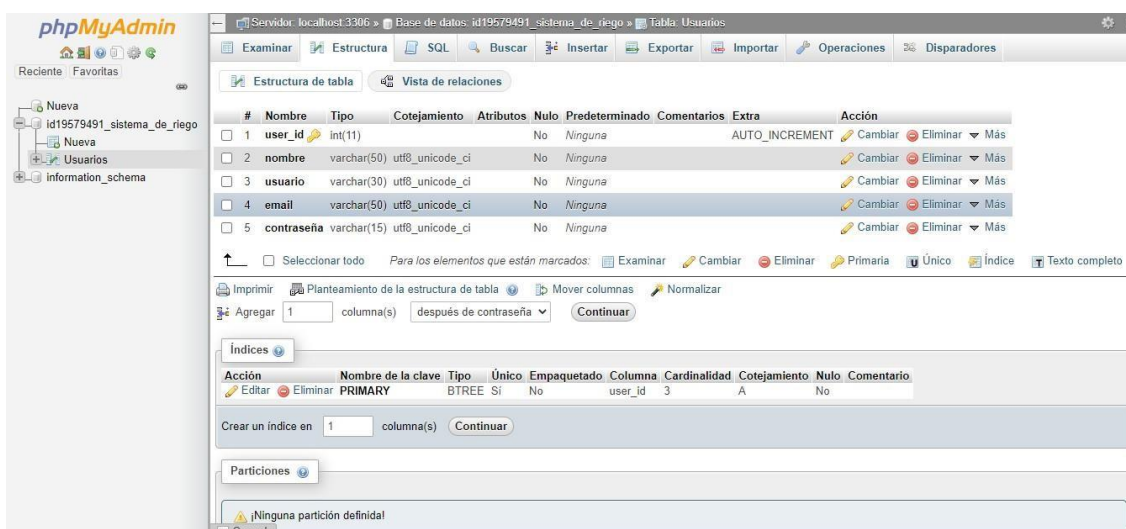


Figura 74 PhpMyAdmin



3.2. Discusión

A partir de los resultados obtenidos, aceptamos la hipótesis general que establece que con el diseño y elaboración de un sistema Arduino de riego automatizado se lograra un uso efectivo del agua.

- Según Perea (2016), confirma que es posible programar el establecimiento de un sistema de riego automatizado, ello es acorde con lo que en este estudio se halla. Por lo que estos resultado guardan relación con lo que define.
- El prototipo de tecnología Arduino que se propone para el riego automatizado controlará el manejo y ahorro de agua, en el Fundo la peña, de la Universidad Nacional Pedro Ruiz Gallo, este prototipo nos permitirá el control y monitoreo de ahorro de agua en los cultivos de pastizales, mediante un software en Android, se visualizará los datos procesados, los componentes o módulos empleados al prototipo Arduino como los Led, lector sd, Relé, etc, se incluye de acuerdo a necesidades y requerimientos que el riego automatizado necesite o establece el usuario y la utilización del lenguaje Arduino.
- Según Tufiño (2018), al implementarse el sistema automático para el control de riego por goteo se pudieron observar muchas mejoras y beneficios obtenidos, lo cual permite que en la producción de cultivos tenga mayores ingresos económicos, lo que confirma la investigación realizada por Tufiño donde quedó demostrado que la producción del cultivo en el grupo experimental, en promedio mejoro un 33.25% con respecto al grupo de control, teniendo en cuenta un 28.61% de mejora de calidad del fruto y el 37.89% de la mejora global de planta.

- Según confirma Moreno (2014), en el prototipo se pudieron obtener resultados positivos para mediciones de humedad del aire ambiente, temperatura y humedad del suelo. Adicionalmente, se pudo verificar que el relé prototipo de encendido y apagado funcionó como se esperaba, ya que cuando el porcentaje de humedad del suelo era bajo, el relé se activaba y en caso contrario se desactivaba. Esta activación y desactivación permitiría que el agua pase a través de la mini bomba de agua que riegan las instalaciones. En el prototipo también se pudo determinar la correcta comunicación entre el microcontrolador y el módulo Bluetooth. capaz de enviar datos continuamente al dispositivo móvil.
- El riego por zonas localizadas aumenta los rendimientos, que en caso de los plantones de palma se traduciría en mayor crecimiento. Por otro lado, es importante recalcar que la inversión inicial para instalar el sistema de riego automatizado es razonable ya que, si analizamos el gasto del personal de vivero a mediano y largo plazo, se tendrá un ahorro considerable, aparte de las ventajas ya antes mencionadas. La automatización de los sistemas de riego en viveros permite un buen manejo del recurso hídrico, Disminuye la aplicación de mano de obra y previene el error humano y garantiza una mayor eficiencia.

CAPÍTULO IV: CONCLUSIONES

- Según lo observado se diagnosticó el estado actual en el fundo la peña, de la Universidad nacional Pedro Ruiz Gallo, que la forma del regadío en sus cultivos se sigue ejerciendo de manera tradicional, por tal motivo se elaboró un prototipo de sistema de riego tecnificado, con el fin de controlar el riego de los cultivos haciendo uso de la tecnología Arduino.
- Se diseñó un diagrama actual de los componentes electrónicos en Proteus, simulando el correcto funcionamiento del prototipo con la tecnología Arduino.
- Se elaboró un prototipo con tecnología Arduino que lee los valores del sensor de temperatura DTH11, sensor humedad, sensor de lluvia, sensor ultrasónico del nivel de agua y a través del módulo Bluetooth, envía información en tiempo real al sistema informático, dicho prototipo funciona correctamente.
- Se creó una pequeña base de datos, en el gestor de base de datos MySQL, en lo cual nos permite almacenar y acceder a los datos múltiples de los usuarios que se logean al momento de registrarse en el aplicativo móvil.
- Se elaboró un aplicativo móvil, para el sistema operativo Android, con una interacción fácil de usar, que establezca una comunicación con el prototipo Arduino, ejecutándose en un dispositivo móvil, se logró recepcionar los datos de los sensores, y visualizarlos en la aplicación móvil, y se concluye que la aplicación móvil funciona correctamente y tiene la aceptación por los usuarios, finalmente, ellos recomendarían la aplicación móvil a otros usuarios.

CAPITULO V: RECOMENDACIONES

Como resultado del presente proyecto se plantean las siguientes recomendaciones, con el fin de seguir mejorando el ahorro de agua en los diferentes cultivos de pastizales.

- Se recomienda utilizar este tipo de sistema informático, para los diferentes tipos de riegos tradicionales lo cual permitirá a hacer un uso eficiente del agua, ya sea para cualquier tipo de cultivo, además también se recomienda seguir con el estudio técnico con tal motivo de poder llevar el sistema a un punto más avanzado, que permita continuar con el monitoreo de los datos de humedad de suelo en tiempo real.
- Para el correcto uso y normal funcionamiento del prototipo, generalmente se recomienda utilizar el programa Proteus para diseñar primero el diagrama esquemático y luego diseñar el circuito electrónico en el Arduino.
- Se debe considerar que, en el futuro, para que los usuarios puedan conectarse al prototipo sin limitación, se recomienda optimizar el módulo bluetooth para una mejor conexión inalámbrica.
- Para los usuarios que deseen utilizar la aplicación y puedan obtener un prototipo de registros de lectura en tiempo real, su información quedará registrada en nuestra base de datos, se recomienda registrar una contraseña que contenga caracteres y números para mejorar la seguridad.
- En el desarrollo de la aplicación móvil se optó por una interfaz básica para el mejor entendimiento del usuario, por lo cual se sugirió una breve capacitación sobre el uso del aplicativo como también del prototipo. Así mismo se consideró hacer un seguimiento y brindar soporte continuo del funcionamiento de la aplicación.

REFERENCIAS

Alegsa, L. (2016). *Sistema Informático (SI)*.

<http://www.alegsa.com.ar/Dic/sistema%20informatico.php>

Alonso, A. (2020). *Agricultura Sostenible*.

<http://blog.pucp.edu.pe/blog/gemrapucp/2020/10/11/agricultura-insostenible/>

Arduino. (2000). *Arduino Home*. <https://www.arduino.cc/en/Main/AboutUs>

Arduino. (s.f.). *Arduino*. <https://www.arduino.cc/>

Barroso, G. A. (2015). *Control y Monitorización de un Invernadero a través de una Aplicación Móvil*. [Tesis de maestría]. E.T.S.I. Diseño Industrial (UPM).

Bellver. (2018). *Tipos de Suelo y sus Características*. <https://tendencias.com/eco/tipos-de-suelos/>

Bustos, D. (2017). *Propuesta de un sistema de control y automatización con administración remota a través de un Smartphone Android para el riego del cultivo de lechuga en la finca los almendros del departamento de Jinotega*. <https://repositorio.unan.edu.ni/8246/>

Castillo, J. C. (2017). *Electrónica*. Editorial Editex, S.A.

Cobo, A. G. (2005). *PHP Y MYSQL*. www.academia.edu/11333117/PHPyMySQL

Crespo, E. (Diciembre de 2014). *Aprendiendo Arduino*.

<https://aprendiendoarduino.wordpress.com/2018/04/14/sensores-arduino-3/>

Developers. (2021). *Android para desarrolladores*. Recuperado de:

<https://developer.android.com/>

Díaz, M. H. (2014). *Prácticas con Arduino*.

www.practicasconarduino.com/manualrapido/entorno_de_programacin.html

EcuREd. (2012). Ecured: Enciclopedia Cubana.

https://www.ecured.cu/EcuRed:Enciclopedia_cubana

Environment, H. (09 de Abril de 2015). *Hidroponia.mx*. <http://hidroponia.mx/que-es-el-riego-tecnificado/>

Escalas, G. (2014). *Diseño y desarrollo de un prototipo de riego automático controlado con Raspberry pi y Arduino*. <https://upcommons.upc.edu/handle/2099.1/25074>

Escobar.C. & Farfán, K. (2018). *Diseño de un sistema de riego para la implementación de cultivos automatizados en el Recinto Playa Seca del Cantón El Triunfo*.

<http://repositorio.ug.edu.ec/handle/redug/36909>

Espinel, M. A. (2012). *Diseño de un sistema de control en tiempo real para el Kernel del sistema operativo*. Asistencia de Tiempo Real.

<http://repositorio.espe.edu.ec/handle/21000/6016>

Ganazhapa, B. O. (2016). *Arduino. Guía Práctica*. http://rclibros.es/wp-content/uploads/2016/04/capitulo_9788494305597.pdf

- Gutierrez.D. (2020). *Automatización de un Sistema de Riego Agrícola por Técnica de Goteo y Aspersión*. <https://www.urp.edu.pe/pdf/id/1929/n/automatizacion-de-un-sistema-de-riego-agricola-por-tecnica-de-goteo-y-aspersion>
- Halfacree, G. (2018). *Raspberry Pi Beginner's Guide*. https://www.raspberrypi.org/magpi-issues/Beginners_Guide_v1.pdf
- ITICITSUR. (03 de Diciembre de 2015). *Introducción a las Tics*. <https://itsuritic.wordpress.com/2015/12/03/las-tics-y-sus-areas-de-aplicacion/>
- Kendall, K. (2007). *Informática de Sistemas*. Rama.
- Labcenter, E. (2022). *PCB Design & Simulation Made Easy*. <https://www.labcenter.com/>
- Lázaro, B. (2008). *Sistema Informático*. https://www.ecured.cu/Sistema_inform%C3%A1tico
- López, R.E. (2015). *Metodologías ágiles de Desarrollo del software aplicadas a la Gestión de Proyectos Empresariales*. Revista Tecnológica. https://www.lareferencia.info/vufind/Record/SV_dffdad893af4284ea4f2f0fdeb4ad406
- Loudon, S. (1999). *Sistema Informático*. Livros Técnicos e Científicos.
- Lujan, J. D. (2019). *Desarrollo de aplicaciones Android con Android Studio: Conoce Android Studio*. RC Libros. <https://books.google.com.pe/books?id=i96LDwAAQBAJ&pg=PA165&dq=android+studio+concepto&hl=es-419&sa=X&ved=0ahUKEwiiu9Oz9rDiAhWIo1kKHVjsCzQQ6AEIJzAA#v=onepage&q=android%20studio%20concepto&f=false>

Moreno, M. C., Silva, G. J., Falcon, C. H., Brito, J. J., & Yeligarcia, Y. G. (2014). *Evaluación del efecto de dos sistemas de riego en el cultivo de la caraota (Phaseolus vulgaris L) en condiciones del Valle de Quíbor, Estado Lara. Irriga. UCLA.*

<https://doi.org/10.15809/irriga.2014v19n4p641>

MECHATRONICS (2013). *Módulo bluetooth inalámbrico*. NAYLAMP MECHATRONICS.

<https://naylampmechatronics.com/inalambrico/24-modulo-bluetooth-hc06.html>

Medina, J. A. (1985). *Riego por goteo*. IICA Biblioteca Venezuela.

https://books.google.com.pe/books?id=sGw2ILQLGg8C&printsec=frontcover&dq=riego+por+goteo&hl=es-419&sa=X&ved=0ahUKEwif-b2_js_1AhUQx1kKHZINCywQ6AEIKDAA#v=onepage&q=riego%20por%20goteo&f=false

MINAGRI. (2015). *Problemas de la agricultura peruana*. MINAGRI PERÚ.

<https://www.minagri.gob.pe/portal/22-sector-agrario/vision-general/190-problemas-en-la-agricultura-peruana?limitstart=0>.

Natsys. (2014). *Todo sobre MySQL: Libro ideal para ingresar en el mundo de la base de datos MySQL*. Editorial Natsys.

<https://books.google.com.pe/books?id=GS3kAgAAQBAJ&printsec=frontcover&dq=mysql&hl=es-419&sa=X&ved=0ahUKEwjvyr2M7oPjAhUxtlkKHZfHA74Q6AEINDAC#v=onepage&q=mysql&f=false>

NOVAGR. (2016). *Electroválvulas de riego*. Novedades Agrícolas S.A.

<https://www.novagric.com/es/electrovalvulas-de-riego>

- ONU (2020). *El agua, un recurso que se agota por el crecimiento de la población y el cambio climático*. <https://news.un.org/es/story/2020/11/1484732?fbclid=IwAR3n-0vH-IQ5Lt27qyJybpP-38HEyviMwa8smAF3ix3sSrQH4MGMpe0opA>.
- Perea, P. (2016). *Diseño de un sistema de monitoreo, registro y control de temperatura y humedad para un cultivo de invernadero*.
<http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/7436/621317P434.pdf>.
- Peiró, A, & López, R. J. (2018). *Arduino: Sensorización y domotización de sistema de riego*.
<https://openaccess.uoc.edu/bitstream/10609/81107/5/rlopez-tarruellaTFG0618memoria.pdf>
- Pomares, J. B. (2009). *Manual de Arduino*.
<https://rua.ua.es/dspace/bitstream/10045/11833/1/arduino.pdf>
- Quispe. (2018). *Diseño de un sistema de riego automatizado por aspersión para viveros de café utilizando la tecnología Arduino*.
http://repositorio.uladech.edu.pe/bitstream/handle/123456789/3024/AUTOMATIZACION_DISENO_QUISPE_TAPARA_HILCIAS.pdf?sequence=1&isAllowed=y
- Renon, M. T. (2013). *Desarrollo de Aplicaciones JAVA*. Editorial Macro EIRL.
<https://books.google.com.pe/books?id=gwkvDgAAQBAJ&pg=PA84&dq=java+en+aplicaciones+m%C3%B3viles&hl=es-419&sa=X&ved=0ahUKEwiKwIDMkM3lAhXDxIkKHZKHANoQ6AEIRzAE#v=onepage&q&f=false>
- Stalling, W. (2000). *Sistemas Operativos*. Editorial Alga Omega S.A.

Suarez, R. C. (2010). *Tecnologías de la Información y la Comunicación*. Editorial España.

https://books.google.com.pe/books?id=oPRegn3QhpgC&printsec=frontcover&dq=tecnologias+en+informacion+y+comunicacion&hl=es-419&sa=X&ved=0ahUKEwqjeKkssDIAhUq1VkKHdqVD_oQ6AEINTAC#v=onepage&q=tecnologias%20en%20informacion%20y%20comunicacion&f=false

Torrente, A. O. (2013). *Curso práctico de formación. ARDUINO*. R.C Libros.

<https://books.google.com.pe/books?id=6cZhDmf7suQC&printsec=frontcover&dq=arduino+curso+practico+de+formacion+oscar+torrente&hl=es-419&sa=X&ved=0ahUKEwjatrub08DIAhWSrVkKHxfIAP4Q6AEIKDAA#v=onepage&q=arduino%20curso%20practico%20de%20formacion%20oscar%20torr>

TOSTATRONIC. (2013). *Sensores de lluvia*. TOSTATRONIC.

<https://tostatronic.com/store/sensores/152-sensor-de-lluvia-yl-83.html>

Tufiño, O. (2018). *Capacitación e implantación de un sistema de riego por goteo para el cultivo de sandía (Citrullus Lanatus) en el colegio Alfredo Pérez Guerrero del Cantón Muisne de la Provincia de Esmeraldas*.

<https://repositorio.unsm.edu.pe/bitstream/11458/3420/1/FISI%20-%20Juan%20Luis%20Tercero%20Sopl%C3%ADn%20Escalante.pdf>

Vázquez, D. G. (2018). *De herramientas digitales en la educación: Redescubriendo el pasado*.

ENID Ed.

<https://books.google.com.pe/books?id=295PDwAAQBAJ&pg=PA1963&dq=caracteristicas+principales+de+las+tic+cabero+1998&hl=es->

[419&sa=X&ved=0ahUKEwjDtISZt8DlAhWK2FkKHZMzCOQQ6AEIKDAA#v=onepag
e&q=caracteristicas%20principales%20de%20las%20tic%20cabero%201998&f=fal](https://books.google.com.pe/books?id=wqAoDwAAQBAJ&source=gbs_navlinks_s)

Villar, E. (2017). *La revolución electrónica*. Editorial Universidad de Cantabria.

https://books.google.com.pe/books?id=wqAoDwAAQBAJ&source=gbs_navlinks_s

ANEXOS

Cuestionario

1. ¿De qué manera es el sistema de riego que emplean en el Fundo La Peña de la Universidad Nacional Pedro Ruiz gallo?

La forma en la que se viene realizando el riego en el fundo, es de manera tradicional, se utilizan unas mangueras de gran longitud, que están conectadas a un tanque con agua (1500 litros), lo cual permite poder regar en las diferentes áreas de cultivos que hay en el fundo. Pero no siempre se llegan a regar todas las parcelas, por lo que en ese caso se hace el uso de baldes para completar el riego.

2. ¿Cuál es la cantidad de agua que se necesita para regar los pastizales?

La cantidad de agua que se necesita para regar los cultivos es de 120 litros, el cual se riega en un tiempo de 10 minutos por parcela. Esto pasa siempre y cuando no haga mucho sol, ya que esto conlleva a que el agua se evapore más rápido y no logre regar lo suficiente a los cultivos.

3. ¿Cada cuánto tiempo se riega los cultivos?

El tiempo que se realiza el riego en los cultivos, va a depender de la cantidad de agua que se utilice en cada etapa de crecimiento de los pastizales, puede ser diario, interdiario o semanal.

4. ¿Cuáles son los tipos de pastizales que existen en el fundo la peña?

Existen la Alfalfa, Moringa, Pasto elefante, Pasto Alemán, Maíz, Lima, Naranja, Algodón, Espárragos.

5. ¿En qué clase de suelo se llega a retener más humedad?

En el suelo Arcilloso o Limoso, este tipo de suelo tiene la capacidad de retener gran cantidad de agua, por lo que los cultivos de pastizales no necesitan ser regados con mucha frecuencia.

8.1 MANUAL DE IMPLEMENTACION

Para exponer de manera específica el desarrollo de nuestra tesis, se realizó este manual de implementación. Con la finalidad de hacer una explicación detallada y breve del software que usaremos a medida se vaya desarrollando el proyecto.

Así mismo detallamos de forma ordenada y progresiva los procesos por etapas a seguir, para la realización de nuestro proyecto sobre el riego tecnificado por goteo.

Por consiguiente, señalamos que este manual de implementación a medida que la tecnología avanza pueda ser actualizado cuando se requiera alguna modificación para la mejora del proyecto.

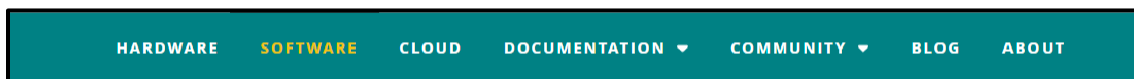
Etapas a realizar en el siguiente manual:

- ETAPA I: INSTALACION DE ARDUINO IDE
- ETAPA II: INSTALACION DE LIBRERÍA DHT-11 EN ARDUINO IDE
- ETAPA III: INSTALACION DE PROTEUS 8 PROFESSIONAL
- ETAPA IV: INSTALACION DE ANDROID STUDIO

8.1.1 ETAPA I: INSTALACION DE SOFTWARE ARDUINO IDE

1. En nuestro navegador predeterminado, nos dirigimos a la página web oficial de ARDUINO o también seleccionado el siguiente link: <https://www.arduino.cc/>
2. Nos dirigimos al menú que está en la parte superior, buscamos la opción de SOFTWARE.

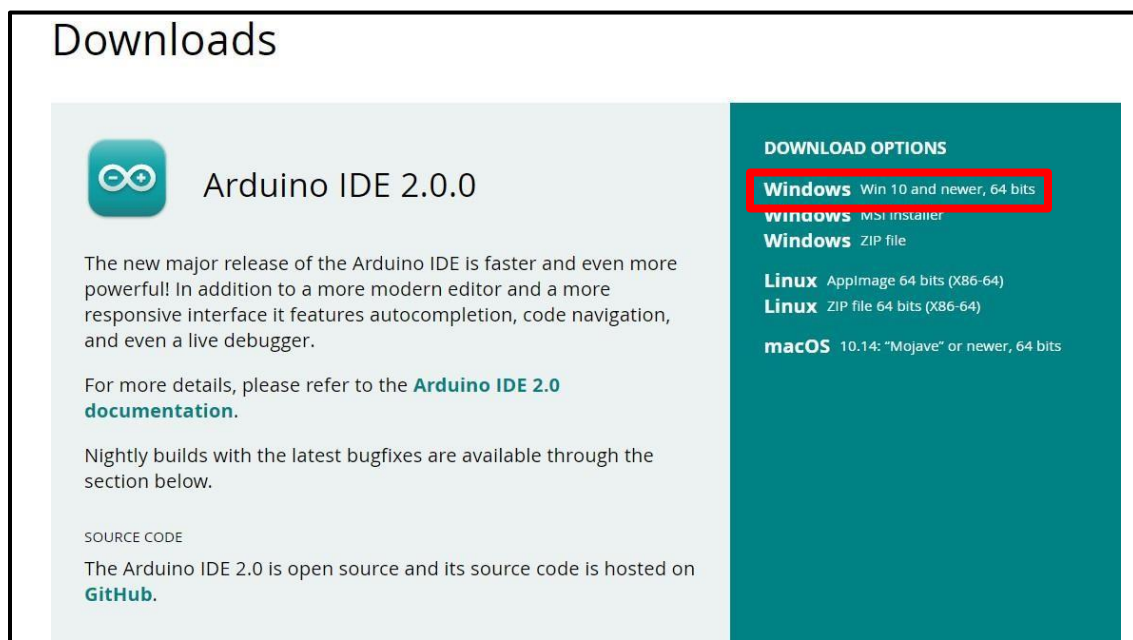
Figura 75 Selección de opción SOFTWARE



Fuente: (Arduino, 2022)

3. Al dar clic nos dirigirá a la sección de Downloads, buscamos la ventana de Arduino IDE 2.0.0 el cual nos mostrará las diferentes opciones de Sistema operativo que pueda ser instalada. En nuestro caso se instaló *Windows win10 and newer, 64 bits*.

Figura 76 Selección de descarga según S.O



Fuente: (Arduino, 2022)

- Una vez seleccionada la opción nos llevara a otra ventana para hacer realizar la descargar.

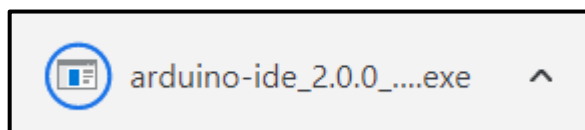
Figura 77 *Descarga de Support the Arduino IDE*



Fuente: (Arduino, 2022)

- Inicio el proceso de descarga del software Arduino IDE.

Figura 78 *Inicio de descarga*



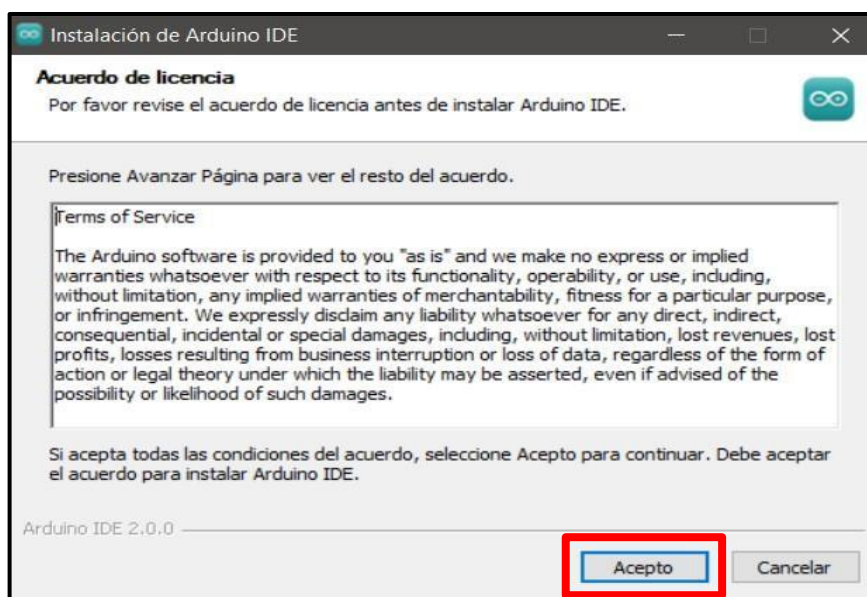
- Al finalizar la descarga y guardada en nuestra unidad de preferencia, procedemos con la instalación haciendo doble clic sobre el ejecutable.

Figura 79 *Ejecutable Arduino IDE*



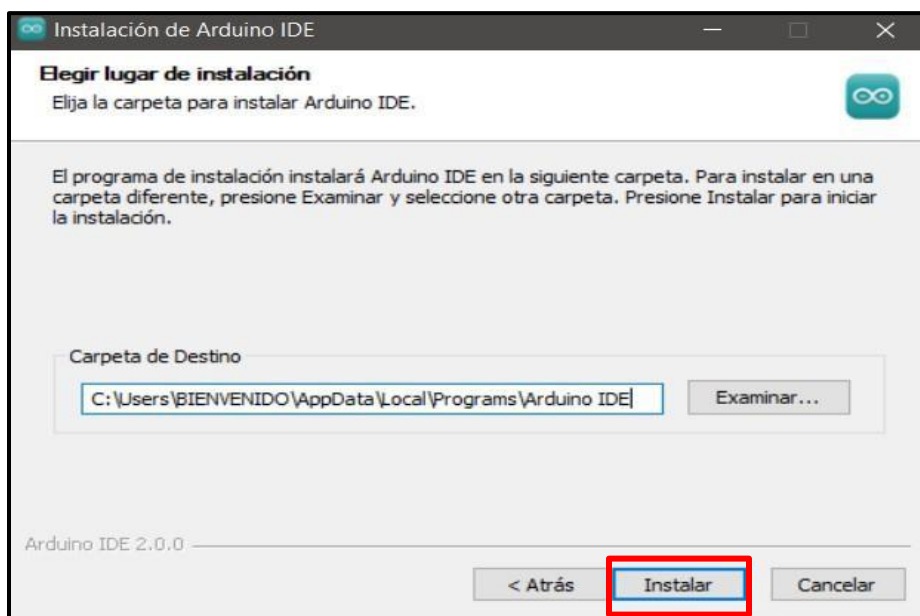
7. Se nos muestra la primera ventana de instalación con las condiciones de acuerdo de licencia, damos clic en Aceptar.

Figura 80 *Acuerdo de licencia*



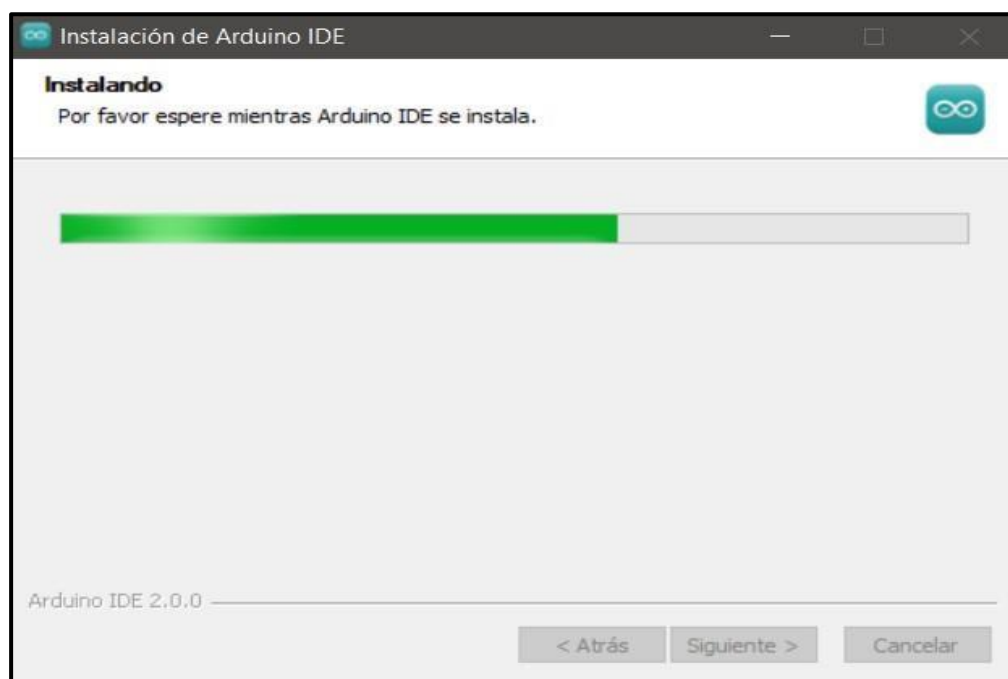
8. La siguiente ventana nos muestra en donde será almacenado nuestro programa, permitiéndonos elegir en que unidad será almacenada, en nuestro se dejó todo por defecto y damos clic en *Instalar*.

Figura 81 *Instalación Arduino IDE*



9. Como se muestra en la siguiente imagen, esperamos que culmine la instalación.

Figura 82 Instalacion paso 2 Arduino IDE



10. Finalizamos la instalación dando clic en *Terminar*.

Figura 83 Instalación paso 3 Arduino IDE



8.1.2 ETAPA II: INSTALACION DE LIBRERÍA DHT-11 EN ARDUINO IDE

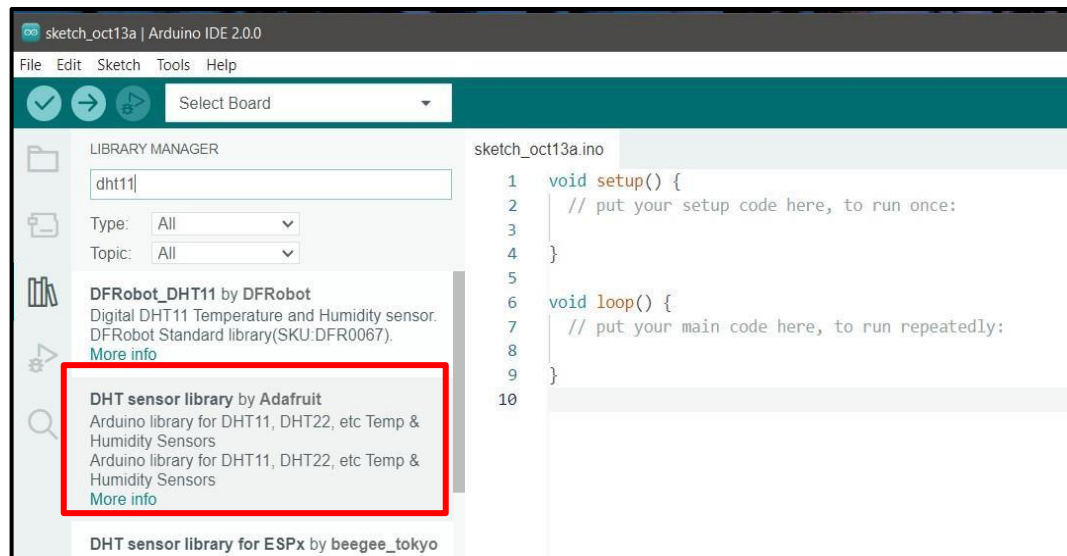
11. Una vez dentro del programa, nos dirigimos al menú de iconos que se encuentra en el lado izquierdo y hacemos clic en el tercer icono (*Library manager*).

Figura 84 Seleccionamos icono



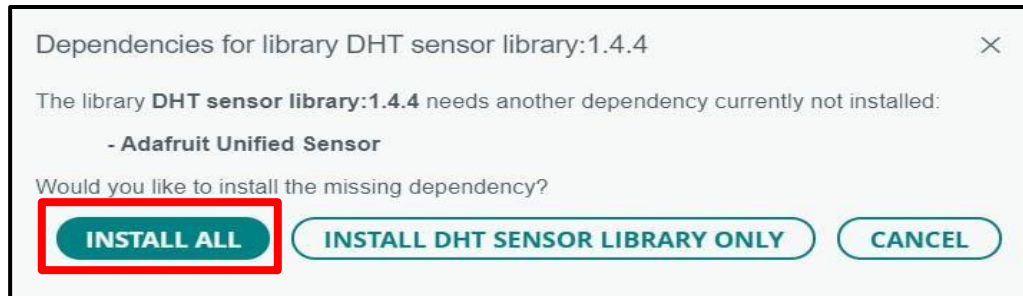
12. En la barra de búsqueda digitamos la librería a usar DHT-11 y seleccionamos la segunda opción para instalar.

Figura 85 Búsqueda de librería



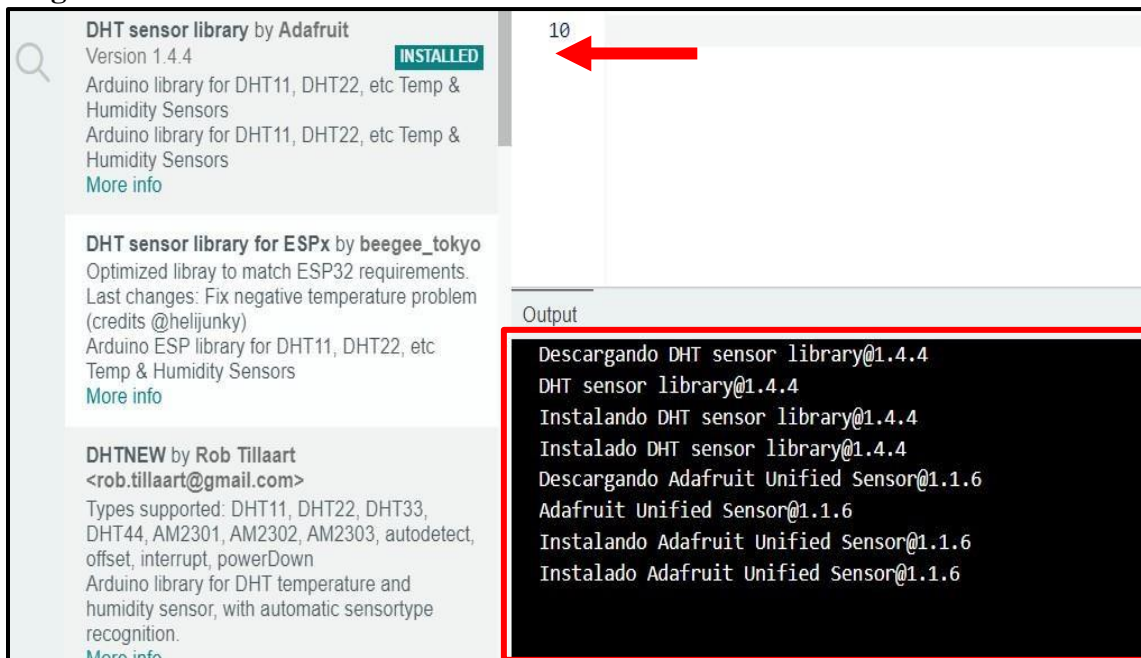
13. Una vez seleccionado la librería a instalar, nos mostrara un mensaje si deseamos instalar todas las dependencias, hacemos clic en *INSTALL ALL*.

Figura 86 *Instalación de librería*



14. En el cuadro inferior derecho nos mostrará la breve instalación y en la barra de búsqueda nos aparecerá la librería ya instalada.

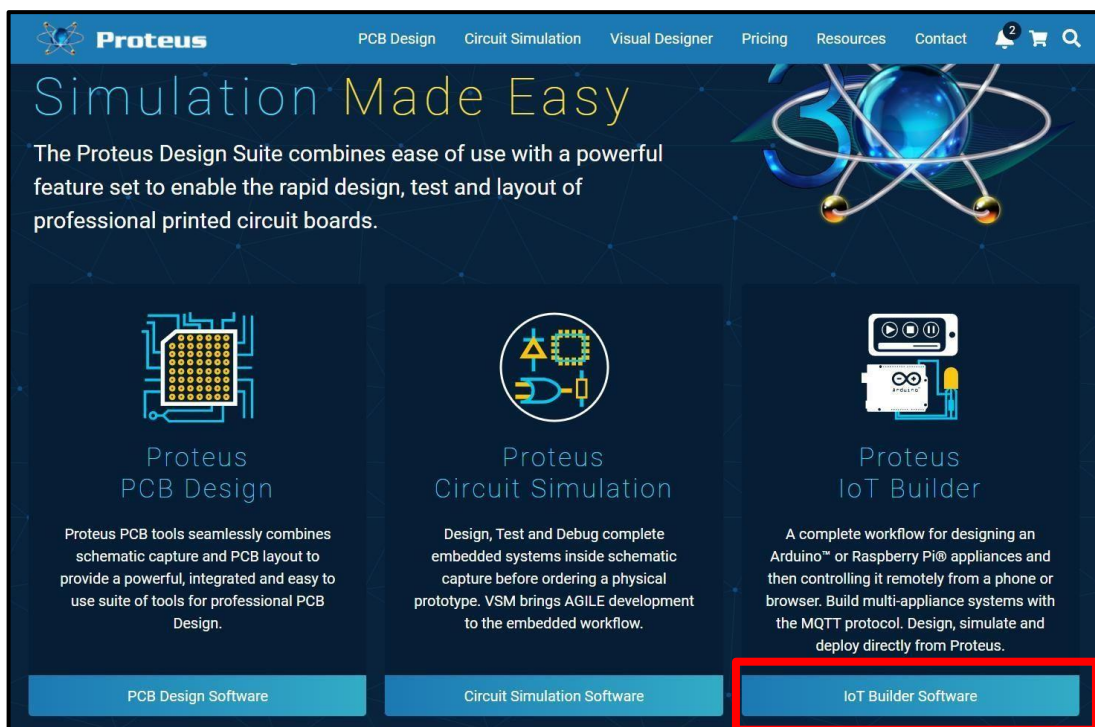
Figura 87 *Librería instalada*



8.1.3 ETAPA III: INSTALACION DE PROTEUS 8 PROFESSIONAL

15. En nuestro buscador predeterminado, nos dirigimos a la página web oficial de Proteus o accediendo por medio de este enlace: <https://www.labcenter.com/>
16. En la página principal, nos centramos en Proteus IoT Builder ya que este trabaja con la tecnología Arduino y damos clic en *IoT Builder Software*.

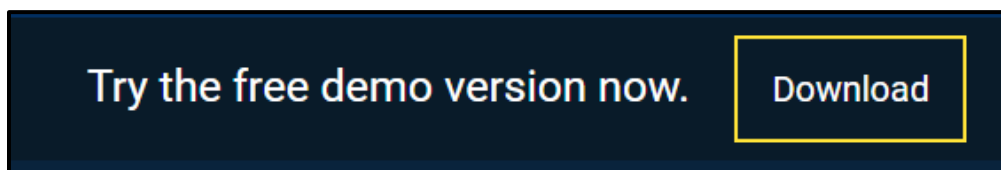
Figura 88 Interfaz principal de Proteus



Fuente: (Labcenter, 2022)

17. En la siguiente ventana nos dirigimos hasta la mitad de la página hasta encontrar la opción de *Download* y damos clic.

Figura 89 Descarga paso 1

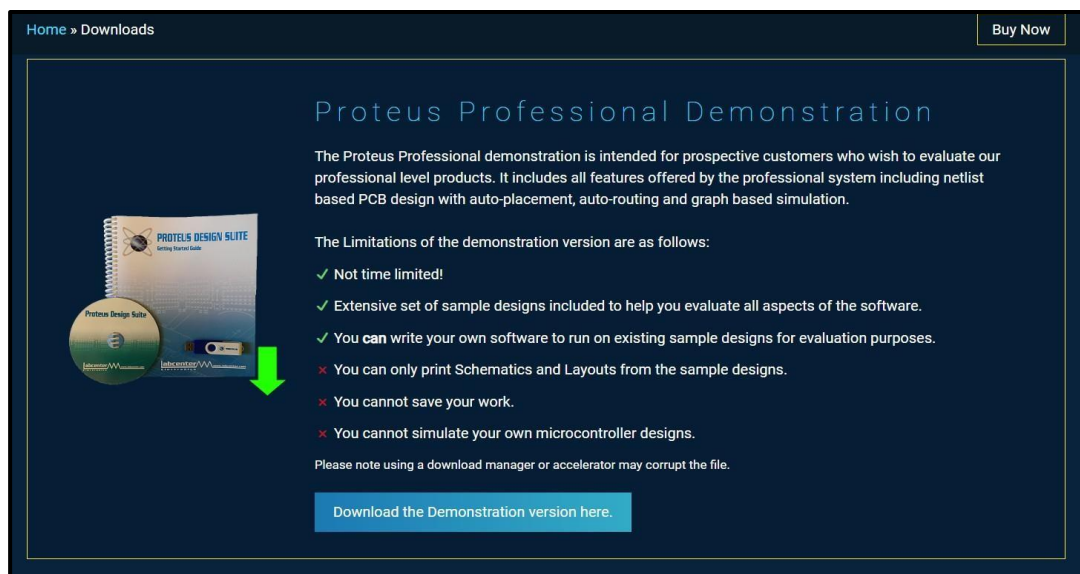


Fuente: (Labcenter, 2022)

Gilberto Carrión Barco

18. Nos direccionara a otra venta de descarga, mismo procedimiento clic en *Download*.

Figura 90 Descarga paso 2



Fuente: (Labcenter, 2022)

19. Nos emergerá una ventana flotante requiriendo nuestro email, en el cual nos llegará el link de descarga, llenamos los campos, la verificación y clic en *Download*.

Figura 91 Verificación de email

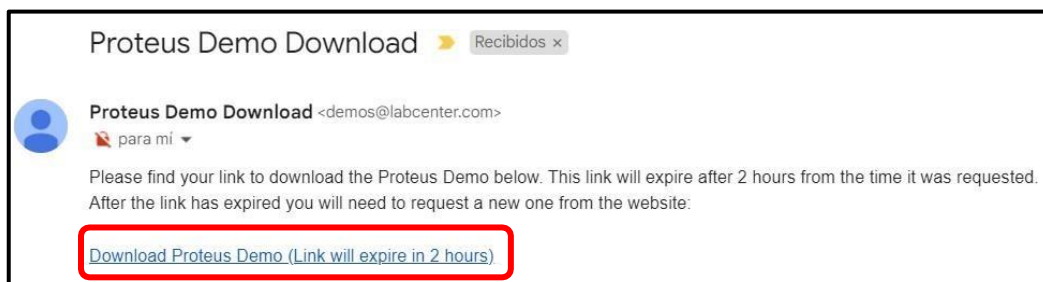
The screenshot shows a floating window titled "Download Proteus Professional Demonstration". It contains a checkbox for "I would like to be contacted by a member of the sales team -". Below this is an "Email Address" field with a red arrow pointing to it. To the right of the email field is a red arrow pointing left. Below the email field, there is a block of text: "An email with a download link will be sent to a valid email. This link will expire after 2 hours. The download is around 600mb. No personal details are stored." Below this text is a reCAPTCHA verification box with a green checkmark and the text "No soy un robot". At the bottom right, there is a blue "Download" button with a red border.


Gilberto Carrión Barco

Fuente: (Labcenter, 2022)

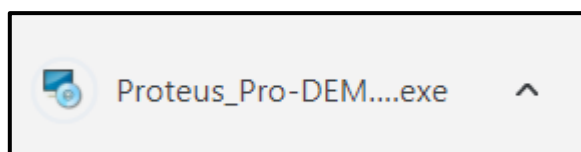
20. Nos dirigimos a nuestro Gmail para verificar el enlace descarga, clic en *Download Proteus Demo*.

Figura 92 *Enlace de descarga*



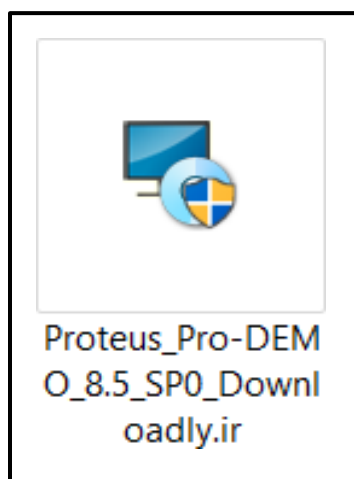
21. Inicia el proceso de descarga de Proteus 8 Professional.

Figura 93 *Inicio de descarga*



22. Al finalizar la descarga, sobre el ejecutable damos doble clic para proceder con la instalación:

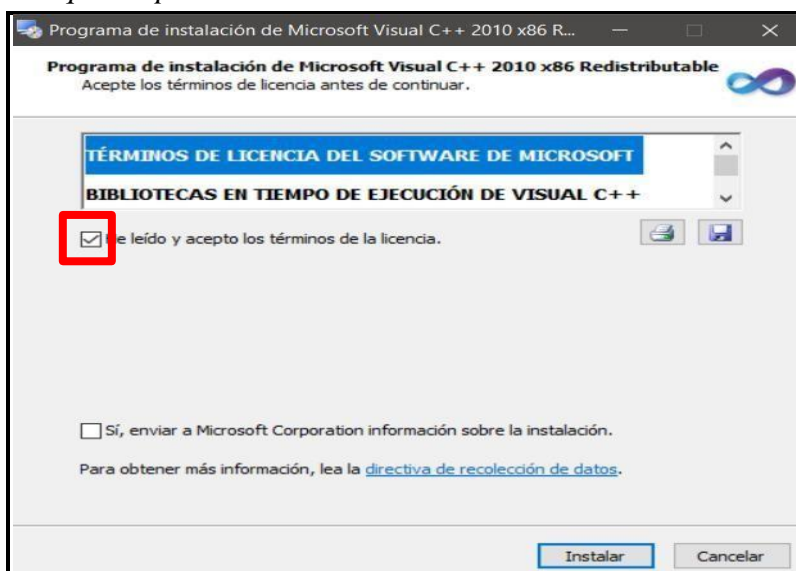
Figura 94 *Ejecutable Proteus 8 Professional*



Gilberto Carrrión Barco

23. La siguiente ventana nos muestra sobre los requisitos que necesita nuestro ordenador para poder continuar con la instalación, marcamos en la casilla vacía de términos y damos clic en *Instalar*.

Figura 95 Primer requisito paso 1



24. Al finalizar la instalación de Microsoft Visual C++ 2010 como requisito, damos clic en *Finalizar*.

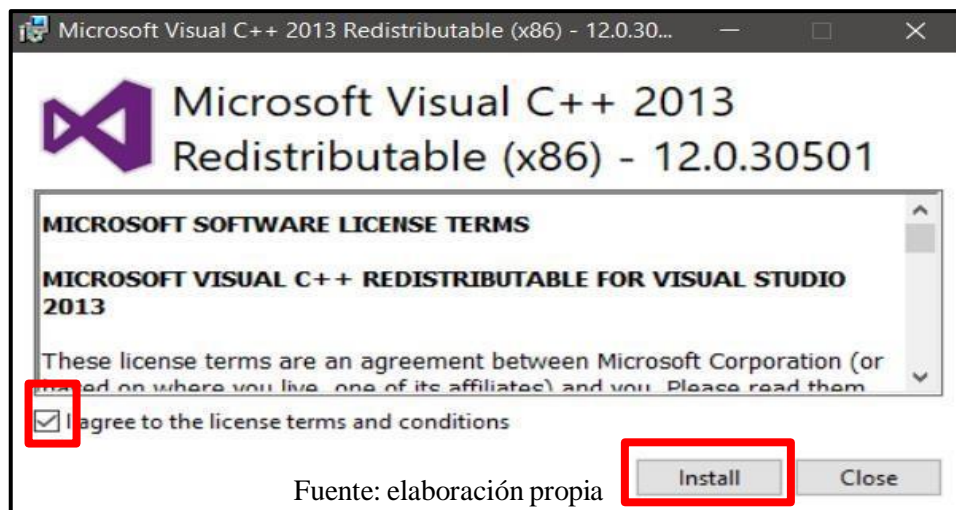
Figura 96 Primer requisito paso 2



Gilberto Carrión Barco

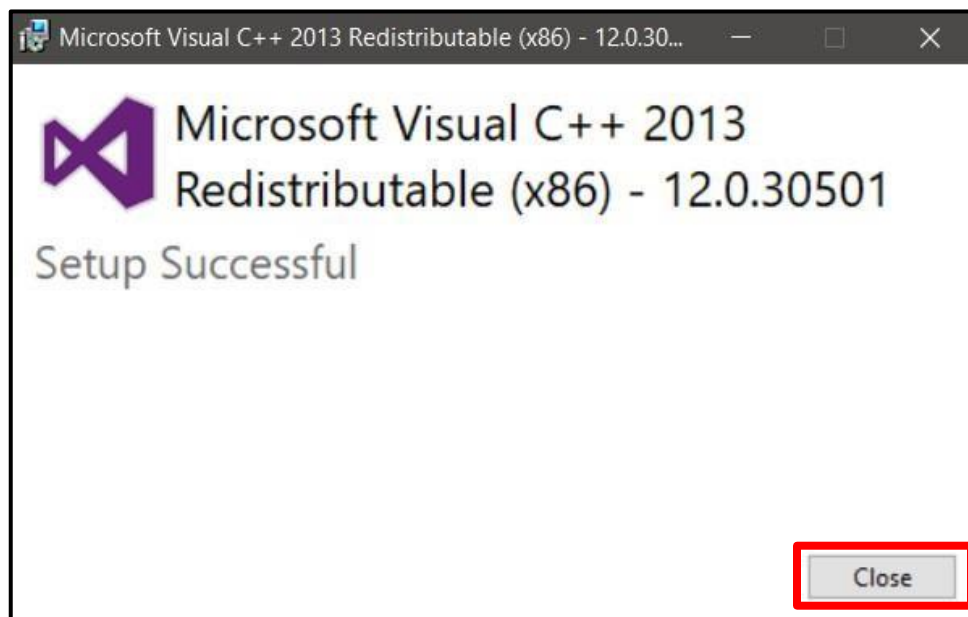
25. El segundo requisito que nos muestra a instalar será Microsoft Visual C++ 2013, marcamos la casilla vacía y clic en *Instalar*.

Figura 97 Segundo requisito paso 1



26. Una vez finalizada la instalación de nuestro segundo requisito damos clic en *Close*.

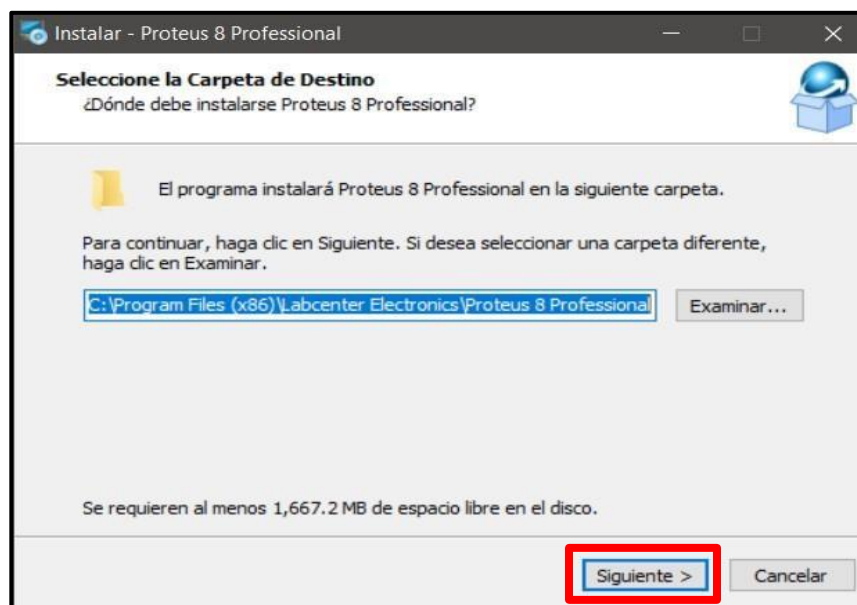
Figura 98 Segundo requisito paso 2



Gilberto Carrión Barco

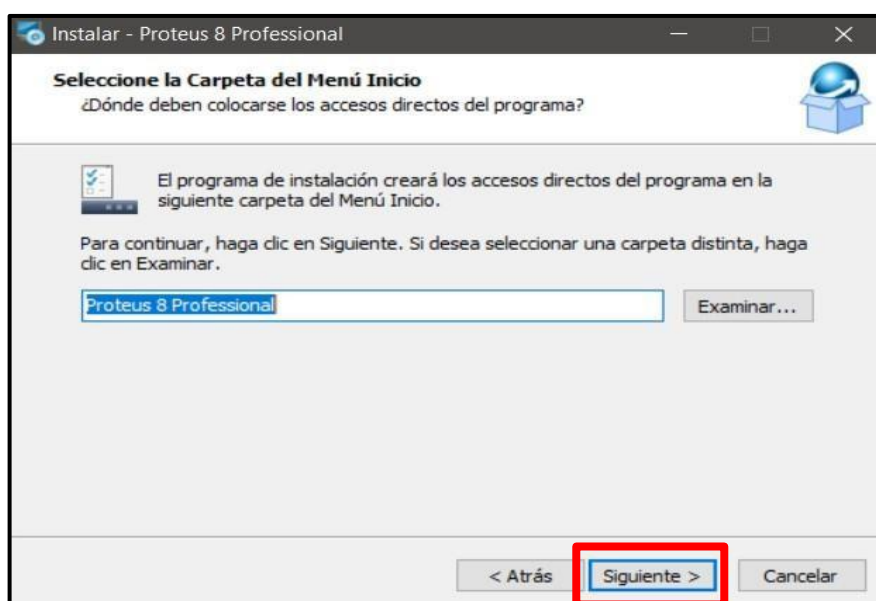
27. Al finalizar las instalaciones de los requisitos, nos mostrará una ventana con la ruta en donde será almacenado nuestro programa, se mantiene todo por defecto y damos clic en *Siguiente*.

Figura 99 *Instalación paso 1 Proteus 8*



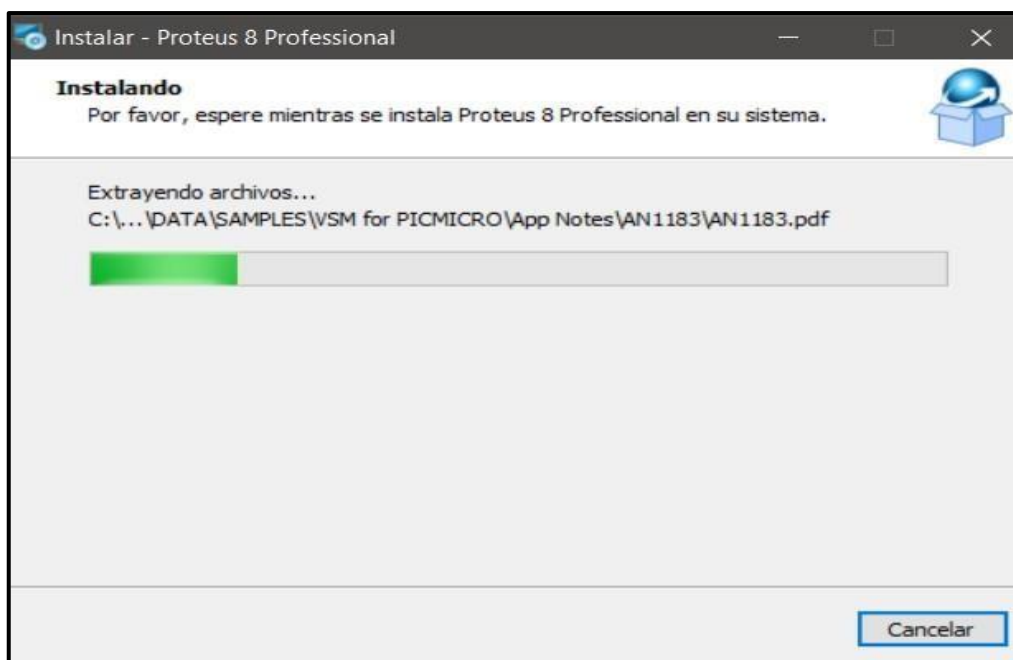
28. Esta ventana nos muestra sobre la creación del acceso directo y el nombre que llevara, igual dejamos todo por defecto, clic en *Siguiente*.

Figura 100 *Instalación paso 2 Proteus 8*



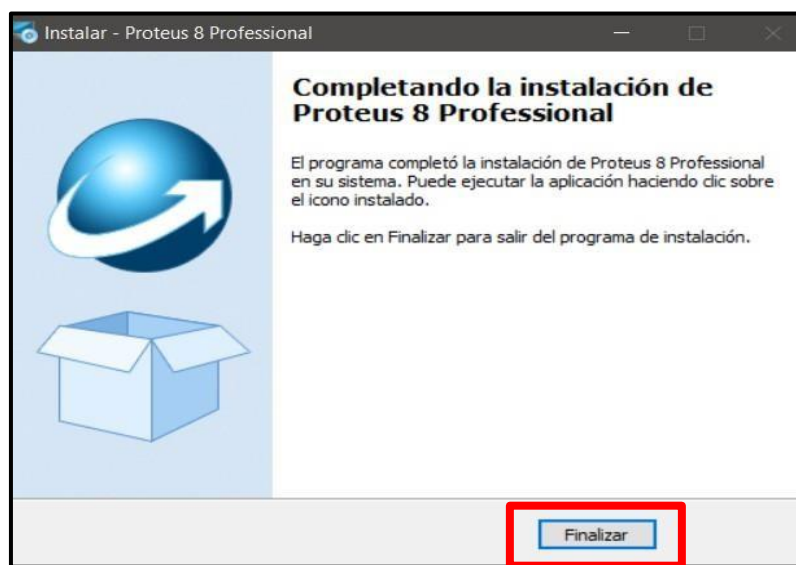
29. Empezara la instalación de nuestro programa, esperamos a que culmine como se muestra en la siguiente imagen:

Figura 101 *Instalación paso 3 Proteus 8*



30. Se termina la instalación dando clic en *Finalizar*.

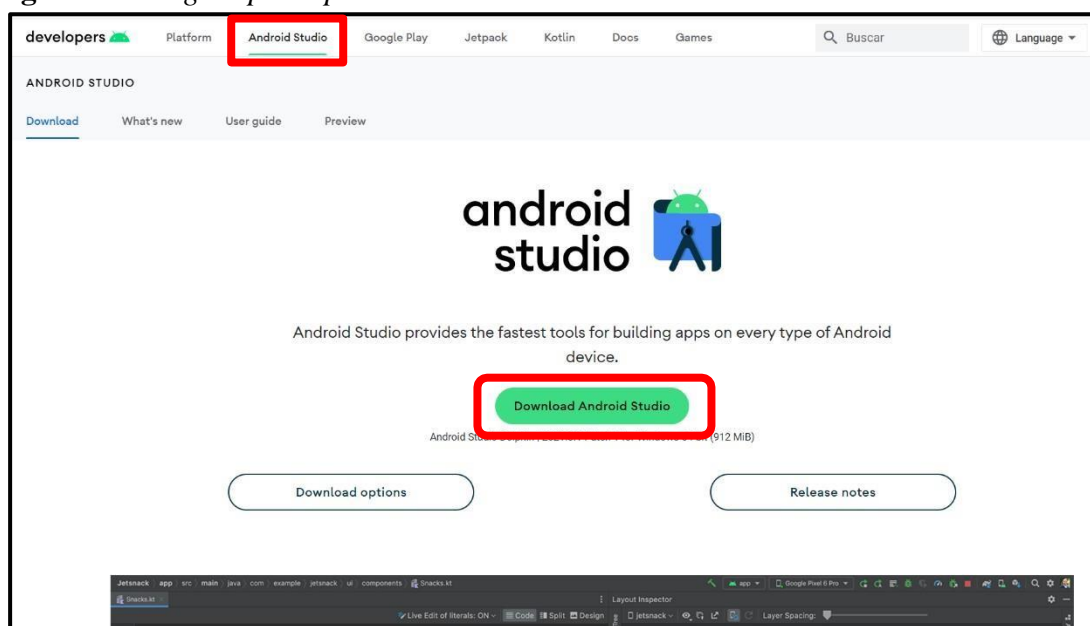
Figura 102 *Instalación paso 4 Proteus 8*



8.1.4 ETAPA IV: INSTALACION DE ANDROID STUDIO

31. En nuestro navegador predeterminado, buscamos la página oficial de Android Studio o también haciendo clic en el siguiente enlace: <https://developer.android.com/>
32. En la página principal, en el menú de opciones que está en la parte superior, damos clic en *Android Studio*, el cual nos llevara a otra ventana de descarga. Donde encontraremos la opción de *Download Android Studio*, clic para ejecutar la descarga.

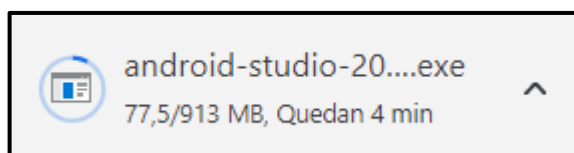
Figura 103 *Página principal de Android Studio*



Fuente: (Developers, 2021)

33. En breve iniciará la descarga, esto dependerá de nuestra conexión a internet.

Figura 104 *Inicio de descarga Android Studio*



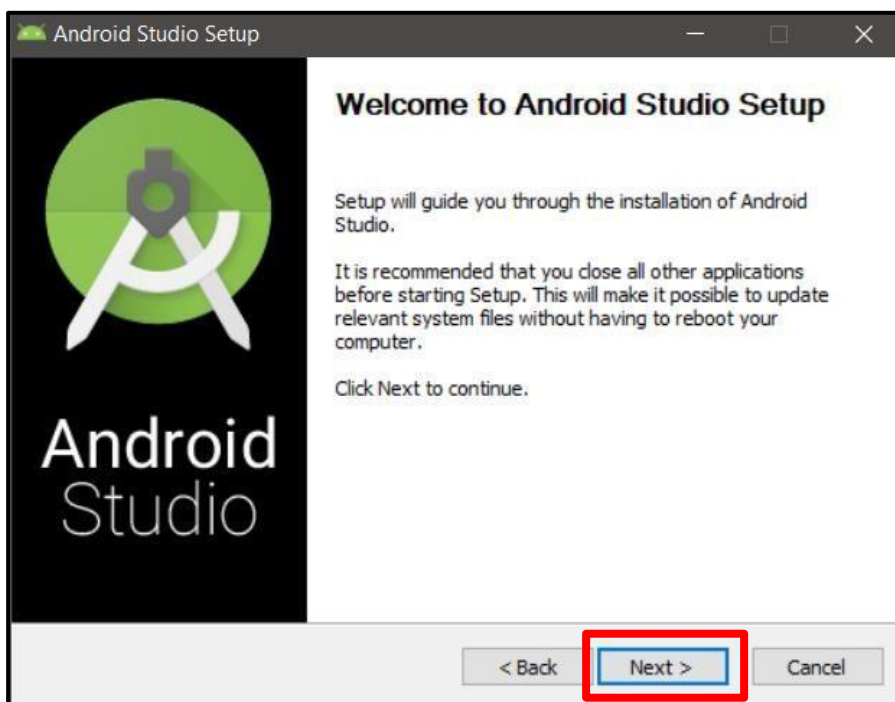
34. Al finalizar la descarga, sobre el ejecutable damos doble clic para proceder con la instalación:

Figura 105 *Ejecutable Android Studio*



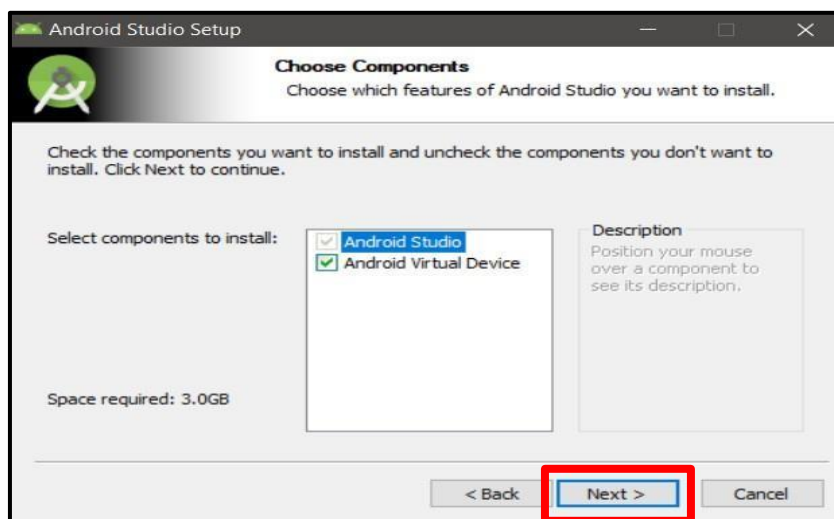
35. Nos muestra la primera venta de instalación y hacemos clic en *Next*.

Figura 106 *Paso 1 Instalación Android Studio*



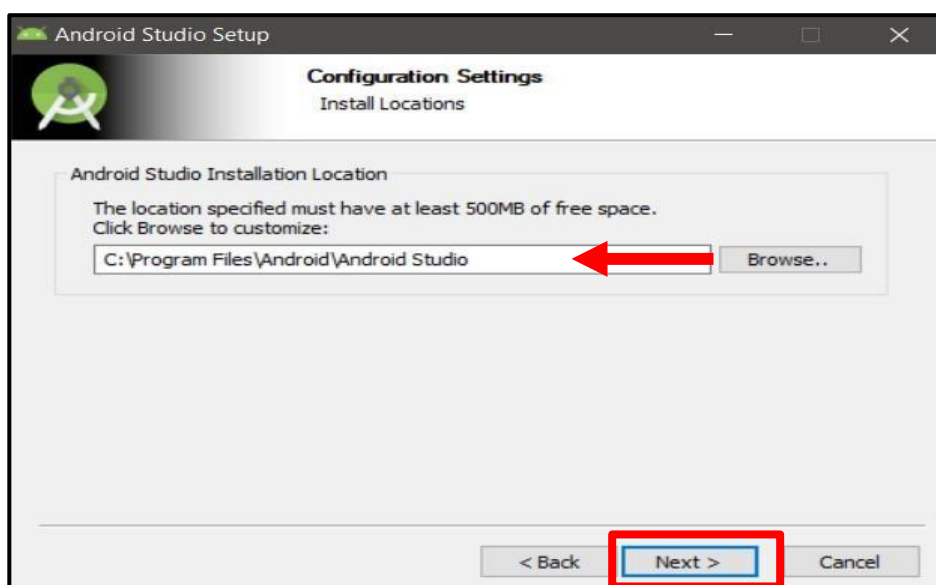
36. En esta ventana nos muestra los componentes que incluye Android Studio, dejamos todo por defecto y clic en *Next*.

Figura 107 Paso 2 Instalación de Android Studio



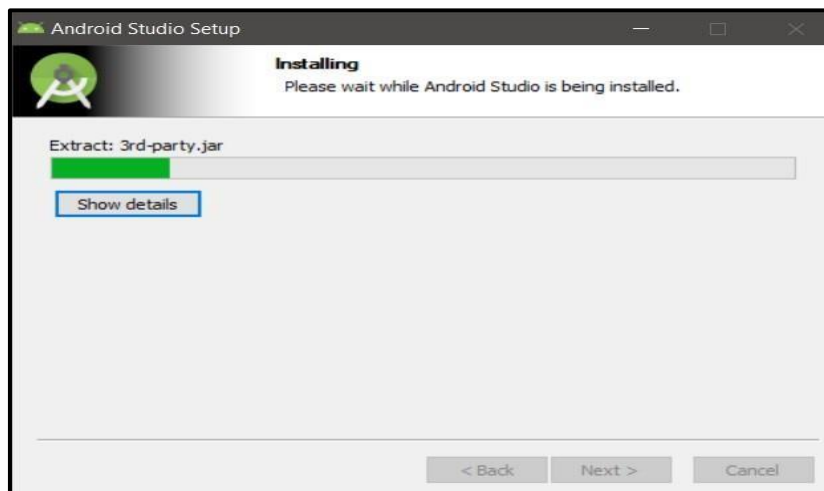
37. La venta siguiente nos indica sobre la ruta en donde se almacenará nuestro programa, para continuar con la instalación dejamos todo por defecto y clic en *Next*.

Figura 108 Paso 3 Instalación de Android Studio



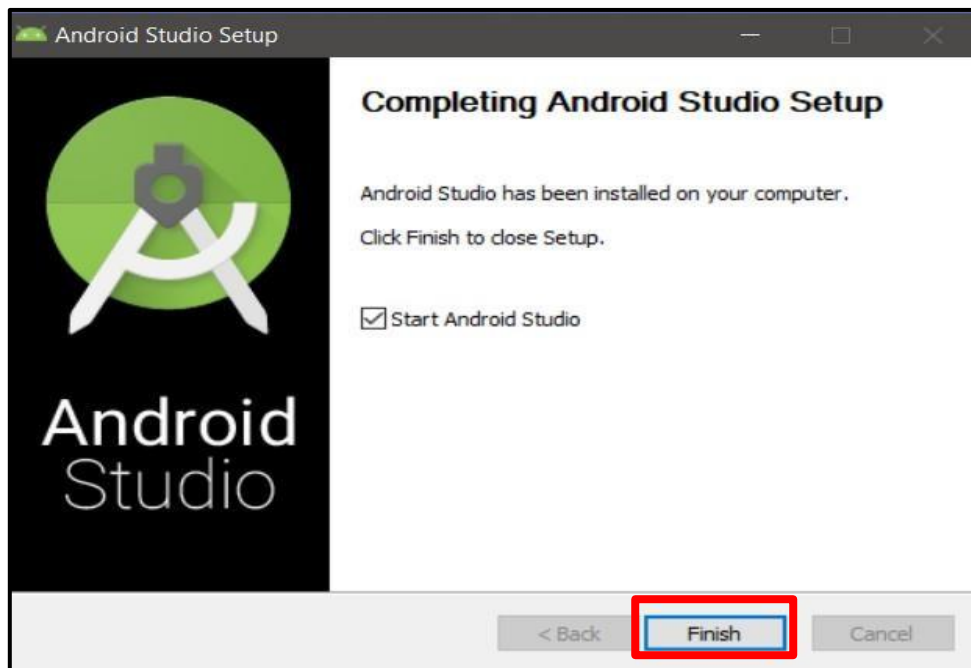
38. Empezara la instalación de los componentes, esperamos a que culmine como se muestra en la siguiente imagen:

Figura 109 Paso 4 Instalación de Android Studio



39. Se culmina la instalación dando clic en *Finish*.

Figura 110 Paso 5 Instalación de Android Studio



CONSTANCIA DE VIRIFICACION DE ORIGINALIDAD

Yo, ***Gilberto Carrión Barco*** usuario revisor del documento titulado **Sistema Informático De Control Y Monitoreo Para Ahorro De Agua En Cultivos De Pastizales En El Fundo La Peña De La Universidad Nacional Pedro Ruiz Gallo.**

Cuyos autores son, **Bach. López Acosta Elmer Arturo y Bach. Vallejos Aguirre Arnold Hidalgo**, identificados con el Documento de Identidad 16720146; declaro que la evaluación realizada por el Programa Informático, ha arrojado un porcentaje de similitud de 19%, verificable en el Resumen de Reporte automatizando solicitudes que se acompaña.

El suscrito analizo dicho reporte y concluyo que cada una de las coincidencias detectadas dentro del porcentaje de similitud permitido no constituyen plagio y que el documento cumple con la integridad científica y con las normas para el uso de citas y referencias establecidas en los protocolos respectivos.

Se cumple con adjuntar el Recibo Digital a efectos de la trazabilidad respectiva del proceso.

Lambayeque, 04 de Septiembre del 2023



Dr. Ing. Gilberto Carrión Barco

16720146

Asesor

Tesis x Turnitin

INFORME DE ORIGINALIDAD

19%

INDICE DE SIMILITUD

18%

FUENTES DE INTERNET

2%

PUBLICACIONES

12%

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

1	hdl.handle.net Fuente de Internet	1%
2	Submitted to Instituto Superior de Artes, Ciencias y Comunicación IACC Trabajo del estudiante	1%
3	repositorio.unap.edu.pe Fuente de Internet	1%
4	repositorio.uladech.edu.pe Fuente de Internet	1%
5	repositorio.unesum.edu.ec Fuente de Internet	1%
6	www.electrogeekshop.com Fuente de Internet	1%
7	es.scribd.com Fuente de Internet	1%
8	tendenzias.com Fuente de Internet	1%
9	www.slideshare.net Fuente de Internet	



Gilberto Carrión Barco

1%

10

www.ichamartin.org

Fuente de Internet

1%

11

prezi.com

Fuente de Internet

1%

12

repositorio.unica.edu.pe

Fuente de Internet

1%

13

repositorio.unprg.edu.pe

Fuente de Internet

1%

14

ri.uaemex.mx

Fuente de Internet

<1%

15

Submitted to Universidad Técnica Nacional de Costa Rica

Trabajo del estudiante

<1%

16

forum.arduino.cc

Fuente de Internet

<1%

17

repositorio.unp.edu.pe

Fuente de Internet

<1%

18

Submitted to Escuela Politecnica Nacional

Trabajo del estudiante

<1%

19

repositorio.utc.edu.ec

Fuente de Internet

<1%

20

www.facebook.com

Fuente de Internet



Gilberto Carrión Barco

<1 %

21

naylampmechatronics.com

Fuente de Internet

<1 %

22

disenoydesarrollodededatos.blogspot.com

Fuente de Internet

<1 %

23

kevinspc.wordpress.com

Fuente de Internet

<1 %

24

Submitted to Universidad Nacional Abierta y a Distancia, UNAD, UNAD

Trabajo del estudiante

<1 %

25

ribuc.ucp.edu.co:8080

Fuente de Internet

<1 %

26

biblioteca.usac.edu.gt

Fuente de Internet

<1 %

27

oa.upm.es

Fuente de Internet

<1 %

28

www.mindmeister.com

Fuente de Internet

<1 %

29

Submitted to National University College - Online

Trabajo del estudiante

<1 %

30

documents.mx

Fuente de Internet

<1 %



Gilberto Carrión Barco

31	repositorio.umsa.bo Fuente de Internet	<1 %
32	Submitted to Unviersidad de Granada Trabajo del estudiante	<1 %
33	blog.pucp.edu.pe Fuente de Internet	<1 %
34	cia.uagraria.edu.ec Fuente de Internet	<1 %
35	repositorio.urp.edu.pe Fuente de Internet	<1 %
36	Submitted to Universidad Pedagogica Trabajo del estudiante	<1 %
37	bivica.org Fuente de Internet	<1 %
38	dspace.unach.edu.ec Fuente de Internet	<1 %
39	eduboticsperu.com Fuente de Internet	<1 %
40	libros.catedu.es Fuente de Internet	<1 %
41	Submitted to Pontificia Universidad Catolica del Ecuador - PUCE Trabajo del estudiante	<1 %



Gilberto Carrión Barco

42

Submitted to Universidad Internacional de la
Rioja

Trabajo del estudiante

<1 %

43

Submitted to Universidad del Atlántico

Trabajo del estudiante

<1 %

44

dokumen.pub

Fuente de Internet

<1 %

45

www.clubensayos.com

Fuente de Internet

<1 %

Excluir citas

Activo

Excluir coincidencias

< 15 words

Excluir bibliografía

Activo



Gilberto Carrión Barco



Recibo digital

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

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Elmer Y Arnold Lopez & Vallejos
Título del ejercicio: Tesis - UNPRG
Título de la entrega: Tesis x Turnitin
Nombre del archivo: TESIS_ARNOLD_TURNITIN_-_24_NOV_1.docx
Tamaño del archivo: 10.62M
Total páginas: 138
Total de palabras: 16,542
Total de caracteres: 91,019
Fecha de entrega: 26-nov.-2022 06:57p. m. (UTC-0500)
Identificador de la entrega... 1963770040

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



TESIS

"SISTEMA INFORMÁTICO DE CONTROL Y MONITOREO PARA
AHORRO DE AGUA EN CULTIVOS DE PASTIZALES EN EL FUNDO LA
PEÑA DE LA UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO"

Presentado para optar el título de

Ingeniero en Computación e Informática

Presentado por:

Bach. Lopez Acosta Elmer Arturo

Bach. Vallejos Aguirre Arnold Hidalgo

Asesor:

Dr. Ing. Gilberto Carrión Barco

Lambayeque - Perú 2022

1

Gilberto Carrión Barco



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS
DECANATO

Ciudad Universitaria – Lambayeque

LICENCIADA - RESOLUCION DEL CONSEJO DIRECTIVO N° 015 -2023-SUNEDU / CD



ACTA DE SUSTENTACIÓN N° 011.-2023.-D/FACFyM

Siendo las 11:00 am. del día Viernes 22 de setiembre del 2023, se reunieron los miembros del jurado evaluador de la Tesis titulada:

" Sistema Informático de control y monitoreo para ahorro de agua en cultivos de pastizales en el fundo: La Peña de la Universidad Nacional Pedro Ruiz Gallo "

Designados por Resolución N° 584-2023-Virtual D/FACFyM de fecha 12 julio de 2023.

Con la finalidad de evaluar y calificar la sustentación de la tesis antes mencionada, conformada por los siguientes docentes:

Dr. Ing. Segundo Pedro Fiestas Rodríguez	Presidente
M.Sc. Ing. Janet del Rosario Aguirre Jalupú	Secretario
Mg. Ing. Carlos Alberto Valdivia Salazar	Vocal

La tesis fue asesorada por (el) (la) Dr. Ing. Gilberto Carrion Barco, nombrado por Resolución N° 883-2019 D/FACFyM de fecha 03 de julio de 2019

El Acto de Sustentación fue autorizado por Resolución N° 787-2023-Virtual D/FACFyM de fecha 18 de setiembre de 2023.

La Tesis fue presentada y sustentada por (el) (los) Bachiller (es): Valleja Aguirre Arnold Hddgo y López Acosta Elmer Arturo y tuvo una duración de 50 minutos.

Después de la sustentación, y absueltas las preguntas y observaciones de los miembros del jurado se procedió a la calificación respectiva, otorgándole el Calificativo de Dieciocho (18) en la escala vigesimal, mención Muy Bueno.

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

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

DR. SEGUNDO PEDRO FIESTAS RODRIGUEZ
Presidente

M. Sc. Ing. Janet del Rosario Aguirre Jalupú
Secretario

MG. ING. CARLOS ALBERTO VALDIVIA SALAZAR
Vocal

DR. GILBERTO CARRION BARCO
Asesor

