



Departamento de Informática
Universidad Técnica Federico Santa María

Universidad Técnica Federico Santa María
Departamento de Informática
Valparaíso - Chile

Trabajo de Titulación para optar al Título de
Ingeniera de Ejecución en Informática

"SISTEMA AUTOMATIZADO PARA RIEGO EN HUERTOS URBANOS Y PLANTAS"

Autora

Rocío Elena Barramuño González

Profesor Guía

Javier Cañas

Profesor Correferente

Hubert Hoffmann

Septiembre - 2021

Ficha del Trabajo Final

Título del Trabajo:	Sistema Automatizado para riego en huertos urbanos y plantas
Nombre de la autora:	Rocío Barramuño González
Nombre del profesor guía:	Javier Cañas
Nombre del profesor correferente:	Hubert Hoffmann
Fecha de entrega (mm/aaaa):	09/2021
Titulación:	Ingeniera de Ejecución en Informática
Idioma del trabajo:	Español
Palabras clave:	Automatización, Optimización, Riego Automatizado, Arduino, Sensores, OpenSource, Agricultura Inteligente, Smart Farming, Sistemas de control de riego, IoT.

AGRADECIMIENTOS

Primeramente agradezco a mi compañero de vida, mi amor, mi familia, Daniel, quien me ha brindado todo su amor, apoyo y me ha acompañado en mis momentos más difíciles, acompañándome y motivándome a seguir adelante.
Agradezco también a las personas que conocí en mi camino, quienes me mostraron otras realidades, me ayudaron a crecer y me enseñaron a valorar la gran oportunidad que es poder estudiar y tener una carrera.
Agradezco a mi profesor guía, quien me apoyó en finalizar esta etapa universitaria.
Y agradezco a mi profesor correferente, quien siempre me escuchó y me apoyó en seguir adelante, pese a todas las dificultades.

RESUMEN

Actualmente en nuestro país se vive una dramática crisis hídrica, viéndose afectado principalmente los sectores agrícolas, productivos y ganaderos [1]. Es por esto que es preciso encontrar rápidas e innovadoras soluciones para hacer frente a este grave problema; que como sociedad no debemos ignorar.
Con el uso de la tecnología, es posible combatir la escasez de este esencial recurso, aportando mejoras en el área de la "Agricultura Inteligente", la cual consiste en la aplicación de soluciones TIC en los procesos de cultivo de la agricultura. Algunas de estas prácticas son, por ejemplo, la agricultura de precisión, el Internet de las Cosas, la utilización de sensores y actuadores, sistemas de geoposicionamiento, vehículos aéreos no tripulados o drones, robots, etc.
Además, se debe destacar que para conseguir una mejora en la gestión del agua en los procesos de cultivo, es necesario analizar tanto las necesidades de cada especie, como las condiciones meteorológicas y ambientales para lograr una distribución eficiente y equitativa del agua.

Centrando el estudio en la sociedad moderna, donde se dispone de escaso tiempo libre y se cuenta con limitados espacios para cultivar; se plantea diseñar un Sistema de Monitoreo y Control Automático para huertos urbanos, que permita automatizar los procesos y tareas en los cultivos mediante el uso de una plataforma de Software de código abierto, Arduino, junto con la utilización de sensores de humedad del suelo y temperatura, para medir las variables y tomar decisiones de riego de acuerdo a los requerimientos de cada planta. De esta manera se optimiza el uso del agua, fundamental para tiempos de escasez hídrica y se obtiene un adecuado desarrollo de los cultivos.
Además, se incluye una aplicación móvil para el registro del cuidado de las plantas, la cual establece un canal de comunicación con el Sistema, mediante el uso de tecnología WiFi.

Los primeros capítulos se enfocan, tanto en la problemática como en las medidas que se tomarán para abordar la solución. Se evaluarán los mercados y productos competidores existentes. Finalizando así, los capítulos con los resultados obtenidos y las futuras mejoras al sistema.

Palabras Clave— Automatización, Optimización, Riego Automatizado, Arduino, Sensores, OpenSource, Agricultura Inteligente, Smart Farming, Sistemas de control de riego, IoT.

ABSTRACT

Currently in our country there is a dramatic water crisis, being affected mainly the agricultural, productive and livestock sectors [1]. This is why it is necessary to find quick and innovative solutions to face this serious problem; that as a society we should not ignore.
With the use of technology, it is possible to combat the scarcity of this essential resource, providing improvements in the area of "Smart Agriculture", which consists of the application of IT solutions in the cultivation processes of agriculture. Some of these practices are, for example, precision agriculture, the Internet of Things, the use of sensors and actuators, geo-positioning systems, unmanned aerial vehicles or drones, robots, etc.
In addition, it should be noted that to achieve an improvement in the management of water in the cultivation processes, it is necessary to analyze both the needs of each species, as well as the meteorological and environmental conditions to achieve an efficient and equitable distribution of water.
Focusing the study on modern society, where there is little free time and there are limited spaces to cultivate; It is proposed to design an Automatic Monitoring and Control System for Urban Orchards, which allows automating the processes and tasks in the crops through the use of an Open Source Software platform, Arduino, with the use of soil moisture and temperature sensors, to measure the variables and make irrigation decisions according to the requirements of each plant. In this way, the use of water is optimized, which is essential for times of water scarcity and an adequate development of the crops is obtained.
In addition, a mobile application is included for the registration of plant care, which establishes a communication channel with the System, through the use of WiFi technology.

The first chapters focus on both the problem and the measures that will be taken to address the solution. Existing competing markets and products will be evaluated.
Thus ending the chapters with the results obtained and future improvements to the system.

Keywords— Automation, Optimization, Automated Irrigation, Arduino, Sensors, OpenSource, Smart Agriculture, Smart Farming, Irrigation control systems, IoT.

INTRODUCCIÓN

Pese a que la pandemia del COVID-19 ha provocado graves problemas de salud a nivel internacional, el confinamiento mundial ha desencadenado efectos positivos en nuestra sociedad, ya que se ha visto un acelerado aumento de actividades de jardinería y horticultura en entornos domésticos. [2]
Estas nuevas actividades resultan ser muy beneficiosas para nuestra salud mental, dado que se ha comprobado que el cuidado de plantas proporciona un efecto sicológico muy positivo en diversas problemáticas como el estrés, la ansiedad y la depresión.

A pesar del interés que existe hoy en día en la jardinería, muchas personas no cuentan con el tiempo, ni con los conocimientos necesarios para dedicarse al correcto cuidado de sus plantas, ya que el mantenimiento de éstas, precisa de diferentes cuidados, como por ejemplo, manejar un correcto riego dependiendo de la especie que se tenga, de la calidad de humedad que hay en el ambiente y de otros factores.
Es por esto que algunas personas terminan deteriorando sus cultivos y/o plantas por exceso o falta de agua, siendo un grave problema para la optimización de este valioso recurso.

En este estudio se diseñará un sistema de riego automático para una óptima gestión del agua en un huerto urbano doméstico. Con este sistema se espera conseguir un riego eficiente y optimizado según las condiciones ambientales y/o meteorológicas, además de las necesidades hídricas de la especie que se tenga. Una adecuada gestión del riego, permitirá el crecimiento óptimo del cultivo, minimizando el consumo de agua.

Puesto que se trata de un sistema doméstico, se ha priorizado el uso de software libre, de bajo costo, simple de adquirir, escalable y de fácil implementación.
Las acciones son realizadas de manera automática por medio de la programación de la tarjeta Arduino, el cual obtiene información del suelo, a través de los sensores de humedad y además obtiene información de la humedad del aire, a través de los sensores de temperatura, estableciendo las decisiones de riego como la apertura y cierre de las electroválvulas de manera automática, para facilitar las labores de riego de agua, manteniendo húmedas ciertas áreas de acuerdo a las necesidades del cultivo.

ÍNDICE DE CONTENIDOS

RESUMEN **4**

ABSTRACT **5**

INTRODUCCIÓN **6**

CAPÍTULO I DESCRIPCIÓN DEL PROYECTO **15**

1.1 Motivación de Investigación 16

1.2 Planteamiento del Problema 16

1.3 Objetivos 16

1.3.1. Objetivo General **16**

1.3.2 Objetivos Específicos **17**

CAPÍTULO II MARCO CONCEPTUAL DEL SISTEMA **18**

2.1 Huerto Urbano 19

2.1.1 Características de un Huerto Urbano **19**

2.1.2 Beneficios de un Huerto Urbano **20**

2.2 Riego 20

2.2.1 Características de un riego tecnificado **20**

2.2.2 Sistemas de Riego **21**

2.2.2.1 Riego por aspersión 22

2.2.2.2 Riego por goteo 23

2.3 ¿Qué es un sistema automatizado? 24

2.3.1 Objetivos de la automatización **25**

2.4 Agricultura inteligente 25

2.5 Indicadores de medición necesarios 27

2.5.1 Temperatura **27**

2.5.2 Humedad del aire **28**

2.5.3 Humedad del suelo **28**

2.5.4 Iluminancia **29**

2.6 Humedad en las Plantas 30

2.6.1 La importancia de la humedad en el crecimiento de las plantas **30**

2.6.2 Requerimiento de agua de los cultivos **31**

2.6.3 Etapas de crecimiento de los cultivos....31

2.6.4 Evaporación....32

2.6.5 Transpiración....32

2.6.6 Evapotranspiración (ET)....33

2.7 Tecnología a utilizar....34

2.7.1 Arduino....34

2.7.1.1 Software y Hardware....35

2.7.1.2 Ventajas....35

2.7.1.3 Bibliotecas....36

2.7.2 Sensores electrónicos....36

2.8 Comunicación entre elementos....36

2.8.1 Protocolo de comunicación 1-Wire....37

2.8.2 Protocolo de comunicación I2C....37

2.8.3 Protocolo de comunicación SPI....39

2.9 Soluciones comerciales existentes....40

2.9.1 Programadores de riego....40

2.9.2 Sistemas inteligentes de riego....41

CAPÍTULO III DISEÑO DEL SISTEMA....42

3.1 Especificaciones y Características del Sistema....43

3.2 Diseño de Hardware....44

3.2.1 Arquitectura del Sistema....44

3.2.2 Especificaciones y Características de los componentes....45

3.2.2.1 Arduino Uno....45

3.2.2.2 Sensor de Temperatura y Humedad del aire - DHT22....47

3.2.2.3 Sensor de Humedad del suelo - Sensor Capacitivo de humedad v1.2....48

3.2.2.4 Sensor de luminosidad BH1750....49

3.2.2.5 Módulo Relé (relay) de 4 Canales....50

3.2.2.6 Bomba de agua sumergible....50

3.2.2.7 Node MCU....51

3.2.2.8 Display LCD 20x4....52

3.3 Diseño de Software 52
3.3.1 Entorno de Desarrollo 52
3.3.2 Algoritmo 54
3.3.3 Aplicación Móvil 57
3.3.4 Librerías utilizadas 60
3.4 Implementación del Sistema 62
3.4.1 Interconexiones 62
3.4.1.1 Sistema Completo 63
3.4.1.2 DHT22 64
3.4.1.3 Sensor Capacitivo de Humedad de Suelo 65
3.4.1.4 Sensor de luminosidad BH1750 66
3.4.1.5 Dispositivo LCD 67
3.4.1.6 Node MCU 68
3.4.1.7 Relé 69
3.4.1.8 Minibomba de Agua Sumergible 70
3.4.2 Prototipo 71
3.5 Presupuesto 73
3.6 Pruebas finales y Resultados 74
3.6.1 Calibración Sensor Capacitivo de Humedad de Suelo 74
3.6.2 Verificación de datos de los sensores 77
3.6.3 Pruebas regulación luminosidad 78
3.6.4 Monitorización de condiciones ambientales 80
3.6.5 Verificación de comunicación entre Arduino y Módulo WiFi 86
3.6.6 Verificación de recepción de datos en la base de datos de Firebase 88
3.6.7 Prueba Activación de riego desde Aplicación Móvil 90
CAPÍTULO IV CONCLUSIONES 94
4.1 Resumen y Conclusiones 95
4.2 Trabajo Futuro 96
4.3 Posibilidad de producto comercial 98
REFERENCIAS BIBLIOGRÁFICAS 99

GLOSARIO **101**

ANEXOS **105**

1. Programa placa Arduino 105
2. Programa placa NodeMCU 109
3. Calibración Sensor de Humedad 112

LISTA DE TABLAS

Tabla N° 1: Ventajas y desventajas del riego por aspersión 22

Tabla N° 2: Ventajas y desventajas del riego por goteo 23

Tabla N° 3: Características de placa Arduino Uno R3 46

Tabla N° 4: Características Generales sensor DHT22 47

Tabla N° 5: Ejemplos de iluminancia 49

Tabla N° 6: Presupuesto Sistema de Riego 73

LISTA DE IMÁGENES

Imagen 1: Huerto urbano en terraza 19
Imagen 2: Tipos de sistemas de riego 21
Imagen 3 y 4: Aspersores fijos y Pivot accionado por motor eléctrico 22
Imagen 5 y 6: Riego por goteo 23
Imagen 7: Estructura de los sistemas automatizados 24
Imagen 8: Esquema del ciclo de la Agricultura de Precisión 26
Imagen 9: Termómetro, sistema de medición de la temperatura 27
Imagen 10: Higrómetro, sistema de medición de la humedad atmosférica 28
Imagen 11: Luxómetro, sistema de medición de la iluminancia 29
Imagen 12: DPV - Déficit de Presión de Vapor 30
Imagen 13: Requerimiento de agua de una planta 31
Imagen 14: Etapas de crecimiento de diferentes cultivos 32
Imagen 15: Proceso de evapotranspiración (ET) 33
Imagen 16: Distintos modelos de placas Arduino 34
Imagen 17: Bus 1-Wire 37
Imagen 18: Bus I²C 38
Imagen 19: Bus SPI 39
Imagen 20 y 21: Programador de grifo y Programador de riego 4 zonas, programación flexible 40
Imagen 22: Diagrama de bloques del sistema de riego automatizado 44
Imagen 23: Placa Arduino Uno R3 45
Imagen 24: Pines placa Arduino Uno R3 46
Imagen 25: DHT22, Sensor de temperatura y humedad ambiental 47
Imagen 26: Sensor Capacitivo de humedad de suelo v1.2 48
Imagen 27: Sensor BH1750 49
Imagen 28: Módulo Relé de 4 canales 50
Imagen 29: Bomba de agua sumergible 50

Imagen 30: Pines placa ESP8266 Node MCU v1.0 51
Imagen 31: Placa ESP8266 Node MCU v1.0 51
Imagen 32: Display LCD 20x4 52
Imagen 33: Entorno de desarrollo IDE Arduino 53
Imagen 34 y 35: Diagrama de flujo Placa Arduino y Diagrama de flujo Módulo WiFi NodeMCU 55
Imagen 36: Configuración Arduino Uno en IDE de Arduino 56
Imagen 37: Configuración módulo WiFi NodeMCU v1.0 en IDE de Arduino 56
Imagen 38 y 39: Aplicación Móvil: Listado de Plantas y Nivel de Iluminancia “Noche" 58
Imagen 40: Aplicación Móvil: Desactivar Riego 59
Imagen 41: Conexión Sistema Completo 63
Imagen 42: Conexión Arduino y Sensor DHT22 64
Imagen 43: Conexión Arduino y Sensor Capacitivo de Humedad de Suelo v1.2 65
Imagen 44: Conexión Arduino y Sensor BH1750 66
Imagen 45: Conexión Arduino y dispositivo LCD 67
Imagen 46: Conexión Arduino y módulo WiFi NodeMCU v1.0 68
Imagen 47: Conexión Arduino y Módulo Relé de 4 canales 69
Imagen 48: Conexión Arduino, Relay, Baterías y Minibomba de Agua Sumergible 70
Imagen 49, 50 y 51: Imágenes del Prototipo 72
Imagen 52: Rango de Calibración, Sensor Capacitivo de Humedad de Suelo 75
Imagen 53: Calibración Sensor Capacitivo de Humedad de Suelo en suelo seco 75
Imagen 54: Calibración Sensor Capacitivo de Humedad de Suelo en suelo húmedo 76
Imagen 55: Calibración Sensor Capacitivo de Humedad de Suelo en suelo encharcado 76
Imagen 56: Prueba recepción de datos desde sensores 77
Imagen 57: Resultados inicialización sensor BH1750 78
Imagen 58 y 59: Resultados medición sensor BH1750 en pantalla LCD 78
Imagen 60: Resultados medición sensor BH1750 en aplicación móvil 79
Imagen 61: Aplicación Móvil: Estado de condiciones medioambientales iniciales 80
Imagen 62: LCD: Estado de condiciones medioambientales iniciales 81

Imagen 63: IDE Arduino: Estado de condiciones medioambientales iniciales 81
Imagen 64: Aplicación Móvil: Prueba Cambio Estado humedad de suelo.....82
Imagen 65 y 66: LCD: Prueba Cambio Estado humedad de suelo. 83
Imagen 67: IDE Arduino: Prueba Cambio Estado humedad de suelo. 83
Imagen 68: Aplicación Móvil: Prueba Cambio Estado temperatura y humedad ambiental.....84
Imagen 69: IDE Arduino: Prueba Cambio Estado temperatura y humedad ambiental. 85
Imagen 70: LCD: Prueba Cambio Estado temperatura y humedad ambiental.....85
Imagen 71: Prueba de conexión módulo WiFi con red WiFi 86
Imagen 72: Prueba de envío de datos desde Arduino al módulo WiFi 87
Imagen 73: Prueba de recepción de datos en módulo WiFi 87
Imagen 74: Prueba de conexión con Firebase.....88
Imagen 75 y 76: Prueba recepción de datos en base de datos de Firebase.....89
Imagen 77: Prueba Activación de riego desde Aplicación Móvil.....90
Imagen 78: Relé activado.....91
Imagen 79: Prueba Desactivación de riego desde Aplicación Móvil 92
Imagen 80: Prueba Relé desactivado 93
Imagen 81 y 82: Ejemplos de caja para protección del sistema 97

CAPÍTULO I

DESCRIPCIÓN DEL PROYECTO

MOTIVACIÓN DE INVESTIGACIÓN
PLANTEAMIENTO DEL PROBLEMA
OBJETIVOS

1.1 Motivación de Investigación

El desarrollo del presente estudio, implica una oportunidad para trabajar con tecnología innovadora, escalable y de bajo costo como lo es Arduino.
A partir de la utilización de este software, es posible ofrecer una solución al problema de escasez hídrica que se está viviendo en nuestro país y a nivel mundial; siendo ésta la principal motivación, ya que permitiría automatizar los procesos dentro de los cultivos y mejoraría la toma de decisiones de riego, con lo cual se optimiza el tiempo invertido en el riego manual de las plantas, y a su vez se genera un uso eficiente del agua.

1.2 Planteamiento del Problema

El problema actual radica en el desperdicio de agua que se genera al realizar el riego de cultivos y/o plantas domésticas de manera manual, debido a que no se cuenta con los conocimientos necesarios sobre los requerimientos hídricos de cada planta y no se toman en cuenta los factores medioambientales, tales como la humedad del suelo y del aire, tan importantes para prevenir el exceso o falta de riego.
A partir de esta problemática, se plantea una solución mediante la elaboración de un sistema automatizado, basado en Arduino, que tiene como objetivo mejorar el riego en huertos urbanos y/o plantas domésticas. Con esta solución se pretende realizar un riego más eficiente, optimizando el uso del agua, y a su vez mejorar la producción de cultivos.

1.3 Objetivos

1.3.1. Objetivo General

El objetivo general es diseñar un sistema de riego automatizado de bajo costo, el cual según las necesidades hídricas de la planta y las condiciones ambientales y/o meteorológicas, permita gestionar de manera autónoma el suministro de agua necesario para conseguir un riego óptimo, minimizando el consumo de este escaso recurso.

1.3.2 Objetivos Específicos

- Estudio y análisis de diversos sistemas de riego para determinar factibilidad en un huerto urbano y/o en plantas domésticas.
- Diseño e implementación del sistema de riego automatizado, de acuerdo a los niveles de humedad para un uso eficiente del agua.
- Desarrollo de aplicación móvil que permitirá interactuar con el sistema y actuar como interfaz de interacción para el usuario, aportando con información adicional de las plantas.

CAPÍTULO II

MARCO CONCEPTUAL DEL SISTEMA

HUERTO URBANO
SISTEMAS DE RIEGO
AGRICULTURA INTELIGENTE
HUMEDAD EN LAS PLANTAS
ARDUINO

2.1 Huerto Urbano

2.1.1 Características de un Huerto Urbano

Un huerto urbano es un espacio al aire libre o de interior destinado al cultivo de verduras, hortalizas, frutas, legumbres, plantas aromáticas o hierbas medicinales, entre otras variedades, a escala doméstica. Este tipo de huerto se puede ubicar en un jardín, en una terraza, en un balcón o simplemente al interior del hogar. En cualquiera de estas opciones, se debe elegir un lugar donde las plantas reciban luz natural y directa la mayor parte del día para el correcto desarrollo de los cultivos.
El cultivo se suele realizar en recipientes tales como macetas, mesas de cultivos, jardineras o jardines verticales, en los que se introduce el mayor volumen de sustrato posible.
Para obtener el mayor provecho del huerto urbano, es importante ir adquiriendo los conocimientos necesarios sobre las especies que se cultiven, ya que tendremos que conocer las épocas de cultivo según la especie, los marcos de plantación, la rotación de cultivos, las plagas en el huerto, las asociaciones entre los cultivos, que tipo de sustrato se adapta mejor a cada especie, las necesidades de riego, etc. para poder obtener un óptimo desarrollo de las especies cultivadas. De entre todas estas características, este proyecto se centrará en la optimización del sistema de riego.



Imagen 1: Huerto urbano en terraza

2.1.2 Beneficios de un Huerto Urbano

- Permite generar alimentos más sanos, nutritivos y libre de pesticidas.
- Mejora la seguridad[1] y soberanía[2] alimentaria.
- Fomenta el autoconsumo[3] de los alimentos que uno mismo cultiva y cosecha, adquiriendo hábitos alimenticios más saludables.
- Promueve una mejor salud mental, combatiendo el estrés, la ansiedad y la depresión.
- Mejora el nivel de actividad física.
- Contribuye a mejorar la calidad del aire.
- Reduce el efecto "islas de calor"[4].
- Fomenta el contacto con la naturaleza y el respeto por ella.
- Aumenta el conocimiento de los alimentos, producción y cosecha.
- Se reduce la huella de carbono, al generar alimentos locales.
- Ayuda a mantener las funciones cognitivas de adultos mayores.
- Fomenta el trabajo en equipo, el desarrollo emocional y social.

2.2 Riego

El riego constituye diversos procedimientos, que permiten la distribución eficiente del agua sobre la superficie del suelo, para satisfacer las necesidades hídricas de los cultivos que no fueron cubiertos mediante la precipitación. Se utiliza en la agricultura y en jardinería.
Actualmente existen distintos tipos de riego, facilitando al agricultor compensar el déficit de precipitaciones y los suministros necesarios para el crecimiento de las plantas.

2.2.1 Características de un riego tecnificado

El riego tecnificado es una práctica utilizada en los cultivos intensivos en la producción agrícola a nivel mundial. Hoy en día es uno de los ejes principales para el desarrollo de este sector económico, ya que se ha convertido en una innovación tecnológica que permite el aprovechamiento óptimo del agua. Dicho factor estratégico permite el ahorro de este recurso natural para el uso agrícola y una mayor productividad, combinando la aplicación de agua, fertilizantes y nutrientes, en la cantidad precisa, de forma localizada y

[1] **Seguridad Alimentaria:** Este concepto hace referencia al acceso a alimentos inocuos y nutritivos para los consumidores en todo momento.
[2] **Soberanía Alimentaria:** Es el derecho de los pueblos a definir y controlar sus sistemas alimentarios y de producción de alimentos tanto a nivel local como nacional, de forma equitativa, soberana y respetuosa con el medio ambiente.
[3] **Agricultura de autoconsumo:** Consumo por parte de los productores de bienes o servicios de los productos que ellos mismos producen.
[4] **Islas de calor:** Aumento de la temperatura por la pavimentación y las edificaciones en las ciudades.

en el momento más oportuno; asimismo, contribuye a controlar todo tipo de plagas y evita el desarrollo de maleza, con lo cual se disminuyen las pérdidas de cultivos.

2.2.2 Sistemas de Riego

De los múltiples sistemas de riego, nos centraremos en los que se pueden aplicar a nuestro proyecto, es decir, en un huerto urbano.

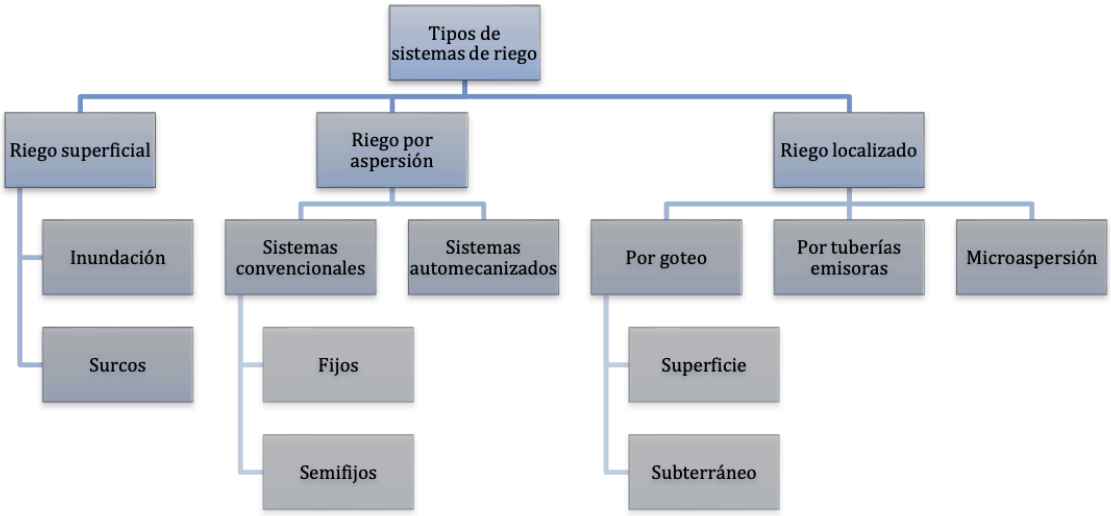


Imagen 2: Tipos de sistemas de riego

2.2.2.1 Riego por aspersión

Este sistema de riego consiste en conducir el agua mediante tuberías hasta los aspersores, los cuales humedecen la tierra de forma similar a como lo haría una lluvia de intensidad y uniformidad variable, con el objetivo de que el agua se infiltre en el punto donde cae. [3] Para esto es necesario una red de distribución que permita que el agua de riego llegue con presión suficiente a los aspersores o difusores.

Los sistemas de riego por aspersión se pueden clasificar en dos grupos generales:

1. Sistemas estacionarios que permanecen en la misma posición mientras dura el riego.
2. Sistemas mecanizados que se desplazan mientras aplican el agua de riego.





Imagen 3 y 4: Aspersores fijos y Pivot accionado por motor eléctrico

Ventajas	Desventajas
- Permite un uso eficiente del agua. - Son capaces de cubrir grandes distancias de terreno. - Se adaptan a cualquier tipo de suelo, sea este plano o inclinado. - Permiten automatizar el riego. - La vida útil de estos sistemas es mucho mayor. - Previene la erosión del suelo.	- La inversión y mano de obra puede ser elevada. - La instalación de estos sistemas es mucho más compleja. - Podría incrementar la aparición de maleza.

Tabla N° 1: Ventajas y desventajas del riego por aspersión

2.2.2.2 Riego por goteo

Es el sistema de riego localizado más popular, el cual permite conducir el agua mediante una red de tuberías y aplicarla a los cultivos a través de emisores que entregan pequeños volúmenes de agua en forma periódica. El agua se aplica en forma de gota por medio de goteros, de esta manera el agua llega directamente a la zona radicular de la planta.
Este tipo de riego, se denomina riego localizado porque únicamente humedece ciertas zonas de volumen de suelo, suficiente para un buen desarrollo del cultivo.
Además, la posibilidad de efectuar riegos frecuentes permite reducir notoriamente el peligro de stress hídrico, ya que es posible mantener la humedad del suelo a niveles óptimos durante todo el período de cultivo, mejorando las condiciones para el desarrollo de las plantas.





Imagen 5 y 6: Riego por goteo

Ventajas	Desventajas
- Permite un crecimiento adecuado del sistema de raíces. - La eficiencia de riego es muy alta (90 al 95%) - Permite la aplicación de fertilizantes en el agua de riego. Esto garantiza una mayor disponibilidad de nutrientes a la zona radicular. - Se adapta a terrenos rocosos o con pendientes. - Contribuye a facilitar el control de malezas, al humedecer el suelo en forma localizada. - Aunque se esté realizando el riego, pueden realizarse otras labores de cultivo como podas, entutorado y aplicación agroquímicos. Actividades que no pueden realizarse simultáneamente cuando se utiliza el riego por aspersión.	- El sistema de goteo puede taparse si no se filtra el agua correctamente. - Es indispensable contar con mano de obra especializada. - Los costos de instalación y diseño son elevados. - Es necesario una limpieza periódica del sistema, tanto en la zona del cabezal como en tuberías y laterales.

Tabla N° 2: Ventajas y desventajas del riego por goteo

2.3 ¿Qué es un sistema automatizado?

La automatización puede definirse como un sistema donde se transfieren tareas de producción, realizadas habitualmente por operadores humanos a un conjunto de elementos tecnológicos.

La estructura de un sistema automatizado consta de dos partes principales:

- Parte Operativa: Es la parte que actúa directamente sobre la máquina. Son los elementos que hacen que la máquina se mueva y realice la operación deseada. Los elementos que forman la parte operativa son los accionadores de las máquinas como motores, cilindros, compresores.

- Parte de Mando: Es la parte que tiene como función controlar el sistema. Suele ser un autómata programable, es decir, un sistema tecnológico que funciona sin la necesidad de que un humano lo controle directamente, el cual realiza las órdenes que le son programadas por el operario. En un sistema de fabricación automatizado el autómata programable está en el centro del sistema. Éste debe ser capaz de comunicarse con todos los constituyentes del sistema automatizado.

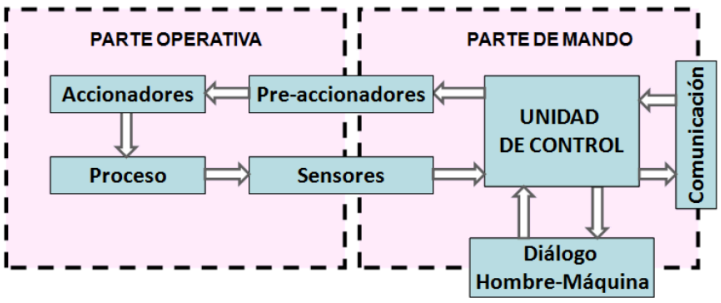


Imagen 7: Estructura de los sistemas automatizados

Los pre-accionadores y los sensores hacen de puente entre la parte operativa y la parte de mando. Los pre-accionadores son los encargados de recibir la orden desde la unidad de control, traducirla y transmitirla a los accionadores para que actúen en el proceso.

2.3.1 Objetivos de la automatización

- Mejorar la productividad, reduciendo los costes de producción y mejorando la calidad de la misma.
- Mejorar las condiciones de trabajo del personal, suprimiendo las tareas repetitivas e incrementando la seguridad.
- Simplificar el mantenimiento de forma que el operario no requiera grandes conocimientos para la manipulación del proceso productivo.

2.4 Agricultura inteligente

La agricultura inteligente o Smart Farming representa la aplicación de Tecnologías de la Información y Comunicación (TIC) en la agricultura, a lo que se ha denominado como la Tercera o incluso Cuarta Revolución Verde. [4]
Internet de las Cosas (IoT), Big Data e Inteligencia Artificial (IA) son algunas de las tecnologías que lideran esta revolución verde [5]. Gracias a estas herramientas es posible obtener datos que permiten mantener la tierra fértil todo el año, automatizar los sistemas de riego, reducir el consumo de agua, controlar los niveles de luz y humedad, además de disminuir el uso de fertilizantes y herbicidas, entre otras múltiples aplicaciones.

El Smart Farming está estrechamente relacionado con tres campos tecnológicos interconectados, que son abordados por la red Smart AKIS[5]:

- Sistema de Gestión de la Información: Sistemas planificados para la recogida, procesamiento, almacenamiento y diseminación de todo tipo de datos necesarios para gestionar las operaciones y funciones de las explotaciones agrícolas.

- Agricultura de Precisión: Es una estrategia de gestión que recoge, procesa y analiza datos temporales, espaciales e individuales y los combina con otras informaciones para respaldar las decisiones de manejo de acuerdo con la variabilidad estimada, y así mejorar la eficiencia en el uso de recursos, la productividad, la calidad, la rentabilidad y la sostenibilidad de la producción agrícola. [6]

[5] **Smart AKIS:** Es una red europea que difunde tecnologías y soluciones de Smart Farming (Agricultura Inteligente) entre los agricultores europeos y que acerca a profesionales de la agricultura, la industria y la investigación para la identificación y desarrollo conjunto de soluciones de Smart Farming en respuesta a las necesidades de los agricultores.



Imagen 8: Esquema del ciclo de la Agricultura de Precisión

- Automatización agrícola y robótica: Consiste en la aplicación de tecnología robótica, inteligencia artificial y control automatizado en todos los niveles de la producción agrícola, incluyendo el uso de robots y drones agrícolas.

La adopción de tecnologías de Smart Farming, facilitan la toma de decisiones y permiten mejoras en la sostenibilidad, la eficiencia de los recursos y la productividad de las cosechas. [7]
Se trata de una transformación digital dirigida no sólo a los grandes productores, sino que también a pequeños agricultores que trabajan predios familiares o están abocados a la agricultura orgánica y que al incorporar estas técnicas pueden potenciar sus emprendimientos para llevarlos al siguiente nivel.

2.5 Indicadores de medición necesarios

Para optimizar el uso del agua en el riego, es necesario analizar distintas variables en tiempo real; algunas de estas variables están asociadas al propio cultivo, como por ejemplo, los requerimientos hídricos de la especie o la fase de crecimiento en que se encuentra la planta, pero otras variables son externas al cultivo y dependen de las condiciones meteorológicas y ambientales; como por ejemplo, la temperatura y la humedad relativa del ambiente, la humedad del suelo, el viento, la radiación solar, etc. Estas magnitudes externas al cultivo son posibles de medir a través de sensores, los cuales entregan la información necesaria para determinar las necesidades hídricas del cultivo analizado.

2.5.1 Temperatura

La temperatura es una magnitud física que refleja la cantidad de calor, ya sea de un cuerpo, de un objeto o del ambiente. La medición de esta magnitud consiste en el uso de un termómetro y se expresa en grados centígrados.
En los cultivos, controlar la temperatura es un factor primordial para obtener un óptimo desarrollo de las plantas, pues influye en sus etapas de crecimiento y desarrollo.
Además, la temperatura es un aspecto importante a considerar en las variaciones de la evaporación del agua, ya que por ejemplo, a mayor temperatura, mayor será la evaporación y por lo tanto, el agua de riego aplicada a la planta se evaporará con mayor facilidad, lo que implicará que la planta reciba una menor cantidad de agua.
Por lo tanto, para obtener buenos rendimientos en las plantas y compensar el efecto de las altas o bajas temperaturas se necesita un óptimo suministro de agua y de nutrientes para que la planta pueda mantener su nivel de metabolismo.



Imagen 9: Termómetro, sistema de medición de la temperatura

2.5.2 Humedad del aire

La humedad del aire ocurre por el vapor de agua que se encuentra presente en la atmósfera. Este vapor procede de la evaporación del agua existente en el planeta, desde mares, ríos hasta plantas o seres vivos.
La temperatura está estrechamente relacionada con la humedad del aire, ya que la cantidad de vapor de agua que puede absorber el aire depende de su temperatura, por lo que a mayor temperatura, el aire admitirá una mayor cantidad de vapor de agua y en consecuencia habrá mayor humedad. Lo mismo ocurre en el caso de bajas temperaturas, ya que el aire frío admitirá una menor cantidad de vapor de agua y esto hará que haya menor humedad de aire.
Una forma de medir la humedad atmosférica es mediante el higrómetro y esta medición se expresa en términos de porcentaje, siendo el tanto por ciento [%] el cociente entre la humedad que contiene el aire y la humedad de saturación.



Imagen 10: Higrómetro, sistema de medición de la humedad atmosférica

2.5.3 Humedad del suelo

La humedad del suelo es la cantidad de agua por volumen de tierra existente en un terreno y es de vital importancia en el mundo de la agricultura, ya que el agua presente en el suelo constituye un factor determinante en la formación, conservación, fertilidad y productividad, así como para la germinación, crecimiento y desarrollo de las plantas cultivadas.
Por lo tanto, conocer los niveles de humedad del suelo determinan el momento óptimo y eficaz del riego de plantas.

Existen distintas técnicas para la medición de este factor, tales como las mediciones gravimétricas, tensiométricas, atenuación de neutrones, disipación de calor y técnicas dielétricas. Los sensores utilizados generalmente en sistemas automatizados, miden la humedad del suelo por la variación de su conductividad o en el caso de sensores capacitivos por la variación de su capacidad, los cuales no tienen la precisión suficiente para realizar una medición absoluta de la humedad del suelo, pero si se podrá tener una referencia válida de la humedad para poder controlar un sistema automatizado de riego.

2.5.4 Iluminancia

La iluminancia o nivel de iluminación, es la cantidad de flujo luminoso que incide sobre una superficie por unidad de área. Su unidad de medida es el Lux (lx), la cual es equivalente a un Lumen por metro cuadrado. Para la medición de la iluminancia se utiliza un luxómetro.
La luz es un factor imprescindible para el crecimiento y desarrollo de plantas, ya que provee la energía necesaria para que las plantas puedan llevar a cabo la fotosíntesis[6], la cual permite convertir la materia inorgánica a materia orgánica, fundamental para la vida sobre la tierra.
Cuanto mayor es la iluminación, más eficaz es la fotosíntesis y más rápido se puede llegar a desarrollar la planta. Sin embargo, a medida que la intensidad de la luz aumenta, la velocidad de la fotosíntesis alcanza un punto máximo, denominado “punto de saturación de la luz”, en donde la velocidad del proceso físico-químico de la fotosíntesis se vuelve plana. Por otra parte, las plantas que reciben insuficientes niveles de luz tienen menor crecimiento vegetativo, menor floración, se debilitan y producen hojas más pequeñas.



Imagen 11: Luxómetro, sistema de medición de la iluminancia

[6] **fotosíntesis:** Es el proceso físico-químico por el cual las plantas, las algas y las bacterias fotosintéticas utilizan la energía de la luz solar para sintetizar compuestos orgánicos.

2.6 Humedad en las Plantas

La humedad atmosférica es muy importante para todos los seres vivos que habitan el planeta, sobretodo en el proceso de crecimiento de plantas, ya que mantener un nivel adecuado de humedad ambiental, permite que se lleve a cabo la fotosíntesis en las plantas y el correcto crecimiento de éstas.

2.6.1 La importancia de la humedad en el crecimiento de las plantas

Las plantas siempre están ajustando las aberturas de las estomas[7] de las hojas según la humedad del aire y el déficit de presión de vapor (DPV)[8]. Si el DPV es alto, significa que la presión de vapor dentro de la planta es mayor que el aire exterior, generando una condición de baja humedad, por lo que más vapor de agua escapa a través de las estomas y más agua necesita la planta, provocando así estrés hídrico. Este proceso de pérdida de agua a través de las hojas se llama transpiración, que abordaremos en detalle más adelante en este mismo capítulo. Por otro lado, si el DPV es bajo, se genera una condición de alta humedad, por lo que las aberturas de las estomas se cierran para minimizar la pérdida de agua, provocando que las raíces de las plantas no puedan absorber nuevos nutrientes y que la planta consuma una escasa cantidad de agua. [8] Desafortunadamente, esto también significa un ralentizamiento de la fotosíntesis que finalmente, repercute en un menor crecimiento de la planta y un mayor riesgo de que se desarrollen enfermedades y patógenos.
Por lo tanto, las dos funciones principales de las plantas que están estrechamente relacionadas con la humedad en el aire y que afectan el rendimiento del cultivo son la transpiración y la fotosíntesis.

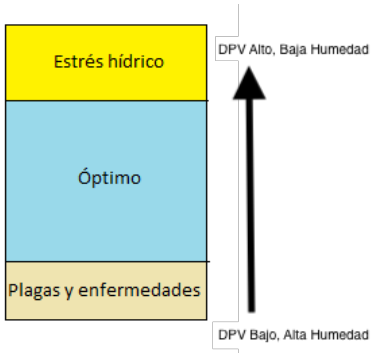


Imagen 12: DPV - Déficit de Presión de Vapor

[7] **Estomas:** Los estomas de las plantas son un tipo celular que permiten el intercambio gaseoso de las hojas de las plantas terrestres. Son uno de los participantes en la fotosíntesis y también en la respiración de las plantas.
[8] **DPV:** Diferencia entre la presión de vapor dentro de la hoja, en comparación con la presión de vapor de aire.

2.6.2 Requerimiento de agua de los cultivos

El requerimiento de agua de los cultivos es la cantidad de agua que la planta necesita a fin de garantizar óptimas condiciones de desarrollo del cultivo.
Este requerimiento se debe en mayor parte a la transpiración, ya que la cantidad de agua utilizada en este proceso es mayor que la almacenada y la utilizada para el proceso de fotosíntesis. Asimismo, durante el crecimiento de los cultivos, el requerimiento de agua va a variar de acuerdo a la etapa de crecimiento en la que estos se encuentren.



Imagen 13: Requerimiento de agua de una planta

2.6.3 Etapas de crecimiento de los cultivos

Comprender las etapas de crecimiento que atraviesan los cultivos es esencial para su correcto cuidado, ya que en cada una de estas fases, las plantas tienen requerimientos distintos en términos de iluminación, humedad, temperatura, espacio, etc. Por ejemplo, una plántula que recién ha germinado necesita una mayor cantidad de humedad que una planta adulta, dado que sus raíces aún se están desarrollando, pero a medida que la planta crece, la humedad se debe ir reduciendo para evitar enfermedades y podredumbre en sus raíces.
En general, a medida que los cultivos se desarrollan, tanto el área del suelo cubierta por la vegetación como la altura del cultivo y el área de cobertura foliar varían progresivamente, debido a las diferencias en evapotranspiración, concepto que abordaremos más adelante en este capítulo.

Estas etapas de crecimiento pueden ser divididos en cuatro fases: inicial, de desarrollo del cultivo, de mediados de temporada y de final de temporada, ilustrados en la siguiente figura.

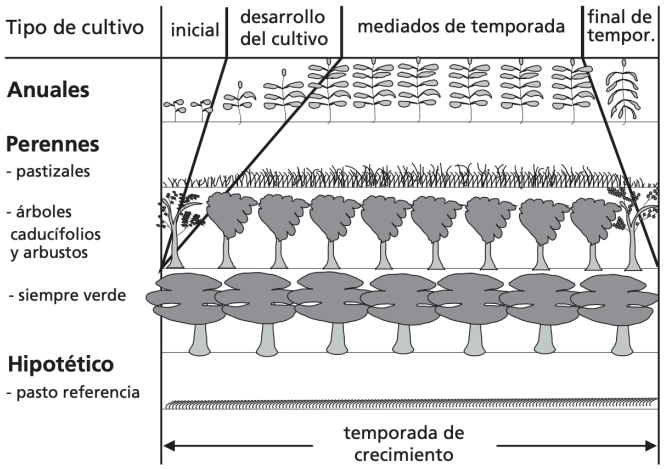


Imagen 14: Etapas de crecimiento de diferentes cultivos

2.6.4 Evaporación

La evaporación es un proceso a través del cual el agua pasa de un estado líquido a un estado gaseoso y vuelve a la atmósfera como vapor de agua (vaporización), gracias a la acción de la energía que llega a la superficie del suelo en forma de radiación solar. [9] Este proceso depende fundamentalmente de la temperatura ambiental, de la radiación solar y otros factores climáticos.

2.6.5 Transpiración

La transpiración es el proceso por el cual el agua pasa de estado líquido a gaseoso, pero a través de los estomas de las plantas, es decir los cultivos pierden agua predominantemente a través de los estomas. El agua y los nutrientes se absorben por la raíz, pero gran parte del agua absorbida se pierde por la transpiración y solo una pequeña parte se queda en los tejidos vegetales.
La transpiración, igual que la evaporación directa, depende del aporte de energía, del gradiente de presión del vapor y de la velocidad del viento. Por lo tanto, la radiación, la temperatura del aire, la humedad atmosférica y el viento también deben ser considerados en su determinación.

2.6.6 Evapotranspiración (ET)

La evapotranspiración es la combinación de dos procesos simultáneos en los que se pierde agua desde la superficie del suelo por evaporación, y desde los cultivos por transpiración. [9]
En las primeras etapas del cultivo, el agua se pierde principalmente por evaporación directa del suelo, pero con el desarrollo del cultivo y finalmente cuando este cubre totalmente el suelo, la transpiración se convierte en el proceso principal.
La evapotranspiración varía en función del clima, las características del cultivo, las prácticas de manejo y el medio de desarrollo, por lo tanto el conocimiento preciso de la ET es una herramienta fundamental que contribuye con una adecuada gestión de los recursos hídricos y con el mejoramiento de la productividad de los cultivos. Esta se expresa normalmente en milímetros (mm) por unidad de tiempo, ya sea hora, día, mes o año. Esta unidad expresa la cantidad de agua perdida de una superficie cultivada en unidades de altura de agua.

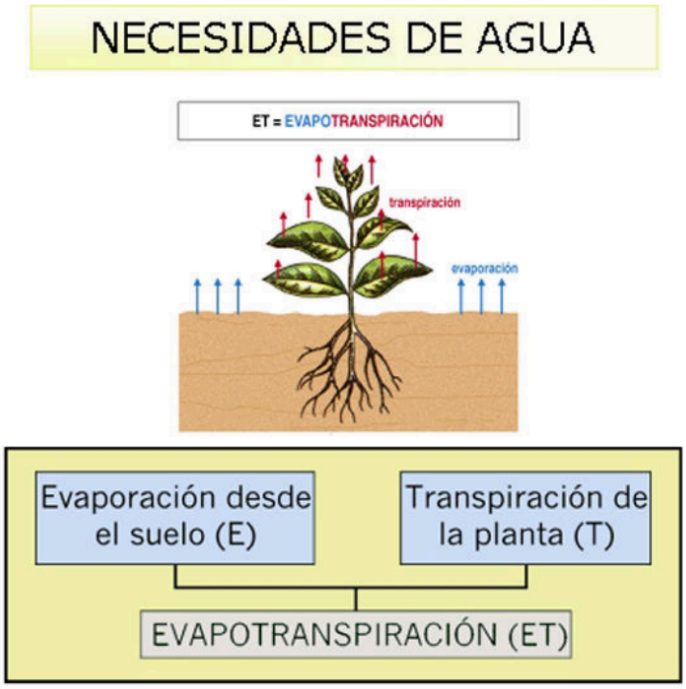


Imagen 15: Proceso de evapotranspiración (ET)

2.7 Tecnología a utilizar

2.7.1 Arduino

Puesto que la propuesta del proyecto es un sistema inteligente, cuyo costo sea el menor posible, se ha considerado utilizar el microcontrolador Arduino.
Arduino, es una plataforma de prototipos electrónicos de código abierto (open-source) basado en hardware y software flexibles, fáciles de usar. Está pensado para estudiantes, diseñadores, y para cualquiera que esté interesado en crear objetos o entornos interactivos. [10]
Arduino puede interpretar el entorno mediante la recepción de sus entradas desde una variedad de sensores y puede afectar su entorno mediante el control de transductores, luces, motores y otros artefactos. El Microcontrolador de la placa se programa usando el "Arduino Programming Language" y el "Arduino Development Environment". Los proyectos de Arduino pueden ser autónomos o se pueden comunicar con software en ejecución en un computador, tablet o equipo móvil.
El software de Arduino se ejecuta en sistemas operativos Windows, Macintosh OSX y GNU/Linux.
Las placas más comunes de Arduino son:

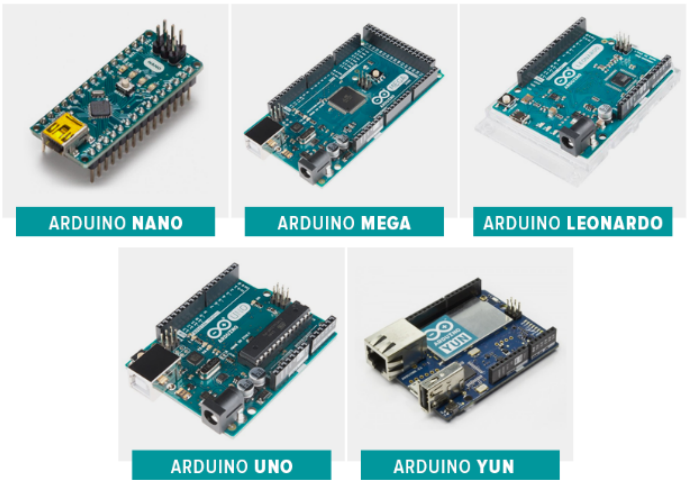


Imagen 16: Distintos modelos de placas Arduino

2.7.1.1 Software y Hardware

La programación de las placas de Arduino se realiza mediante el software de Arduino IDE ("Integrated Development Environment"). Este IDE es una aplicación multiplataforma que podemos instalar en un computador, pudiendo utilizarlo tanto en sistemas Windows como en macOS o Linux. Por medio de este IDE se podrá programar las acciones que queremos que realice nuestro microcontrolador, permitiendo cargar los programas en placas compatibles con Arduino.
La plataforma Arduino se programa mediante el uso de un lenguaje propio ("Arduino Programming Language"), el cual está basado en el lenguaje de programación Processing, que es un lenguaje orientado a diseñadores, fácil de usar y similar al lenguaje C, por lo que soporta todas las funciones del estándar C y algunas de C++, utilizando reglas especiales de estructuración de códigos.

2.7.1.2 Ventajas

- **Software y hardware de código abierto y extensible:** El software Arduino se publica como una herramienta de código abierto, es decir que puede ser usado y modificado por los usuarios sin ninguna restricción. Esto permite que exista un rico ecosistema de placas electrónicas no oficiales para distintos propósitos y de librerías de software de tercero, que pueden adaptarse mejor a nuestras necesidades.
- **Tiene una gran comunidad:** Gracias a su gran alcance hay una gran comunidad trabajando con esta plataforma. Así se genera una cantidad de documentación bastante extensa, la cual abarca casi cualquier necesidad. [10]
- **Su entorno de programación es multiplataforma:** El software Arduino (IDE) se puede instalar y ejecutar en sistemas operativos Windows, Mac OS y Linux. La mayoría de los sistemas de microcontroladores están limitados a Windows.
- **Lenguaje de programación de fácil comprensión:** Su lenguaje de programación basado en C++ es de fácil comprensión. C++ permite una entrada sencilla a los nuevos programadores y a la vez con una capacidad tan grande, que los programadores más avanzados pueden exprimir todo el potencial de su lenguaje y adaptarlo a cualquier situación.
- **Bajo costo:** Las placas Arduino son relativamente baratas en comparación con otras plataformas de microcontroladores.
- **Re-usabilidad y versatilidad:** Es re-utilizable porque una vez terminado el proyecto es muy fácil poder desmontar los componentes externos a la placa y empezar con un nuevo proyecto. [10]

2.7.1.3 Bibliotecas

Para el adecuado funcionamiento de los sensores, actuadores y elementos del sistema de control se hará uso de distintas bibliotecas, las cuales facilitarán la programación, la comunicación entre los componentes y la automatización.
Actualmente existe una gran cantidad de bibliotecas open source, las que podemos añadir fácilmente al IDE y utilizarlas libremente, además el entorno de Arduino ya incluye algunas bibliotecas por defecto que facilitan el desarrollo.

2.7.2 Sensores electrónicos

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, presión, humedad, movimiento, pH, etc.
Acotándonos a nuestro estudio, los sensores nos permitirán obtener los datos de entrada de las diferentes variables del sistema que vamos a evaluar, las que posteriormente serán analizadas por el microcontrolador y así finalmente poder adoptar decisiones oportunas de riego.
Dado que se analizarán las condiciones meteorológicas y ambientales, se utilizarán sensores de temperatura, de humedad del aire, de humedad del suelo e iluminancia.

2.8 Comunicación entre elementos

Para establecer la comunicación de los componentes y sensores con la placa Arduino, se utilizan diferentes protocolos de comunicación. A continuación se nombrarán los protocolos más significativos y cercanos a los sistemas embebidos basados en microcontroladores.

2.8.1 Protocolo de comunicación 1-Wire

1-Wire es un protocolo de comunicaciones en serie que fue diseñado por Dallas Semiconductor y está basado en un bus, un maestro y varios esclavos de una sola línea de datos en la que se alimentan. En otras palabras, el bus 1-Wire consiste en un esquema de señalización, direccionamiento y arbitraje que permite comunicaciones bidireccionales entre un dispositivo maestro y uno o varios periféricos (esclavos) sobre un solo hilo.
Para poder utilizar este bus 1-Wire en Arduino, el IDE proporciona la librería "OneWire.h", que contiene las funciones necesarias para controlar el hardware integrado.

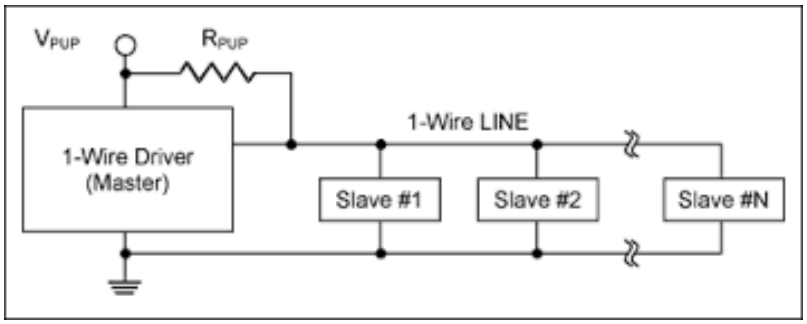


Imagen 17: Bus 1-Wire

Se utilizará este protocolo para el sensor de temperatura y humedad del aire (DHT22).

2.8.2 Protocolo de comunicación I²C

El protocolo I²C (Inter-Circuitos Integrados) es un sistema de comunicación bidireccional basado en 2 hilos, el SDA (Datos) y el SCL (reloj) muy parecido al formato de comunicación Serial. [11]
Este protocolo fue desarrollado a principios de los años ochenta, por Phillips Semiconductors y el propósito original de este protocolo era proveer una manera sencilla de conectar la CPU con los chips periféricos en los televisores y demás sistemas de complejidad media-alta. Hoy por hoy, este bus es muy utilizado en la industria, principalmente para poder comunicar microcontroladores y sus periféricos en sistemas integrados (Sistema Embebido).

El protocolo I²C es síncrono[9] y está diseñado como un bus maestro-esclavo [12]:

- Maestro: Son los que inician y coordinan la comunicación. Usualmente, este es el rol que cumple Arduino en un bus I²C.
- Esclavos: Son los dispositivos que están a la espera de que algún maestro se comunique con ellos. Comúnmente este rol es desempeñado por los sensores y actuadores que soportan este bus I²C, aunque también es posible que un microcontrolador se comporte como un esclavo.

Una de las ventajas que tiene este protocolo es que permite conectar varias tarjetas "Maestras" mediante un modo multimaestro, pudiendo comunicar hasta 128 dispositivos.

Para poder utilizar este bus I²C en Arduino, el IDE proporciona la librería "Wire.h", que contiene las funciones necesarias para controlar el hardware integrado.

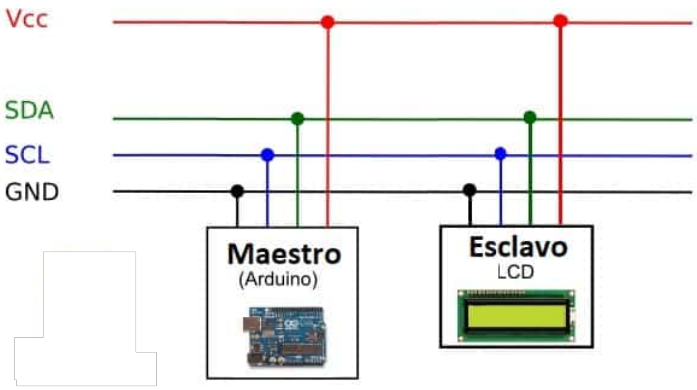


Imagen 18: Bus I²C

Se utilizará este protocolo para la placa de Arduino Uno, para la placa ESP8266 Node MCU, para el dispositivo LCD y para el sensor BH1750.

[9] **Síncrono:** Quiere decir que el envío y recepción de datos está sincronizado con una señal de reloj que comparte con la placa de Arduino, ya que se utilizan dos líneas separadas una para datos y otra para el reloj.

2.8.3 Protocolo de comunicación SPI

El protocolo SPI (Serial Peripheral Interface) es un protocolo de comunicación síncrona de 4 hilos, que proporciona mayor velocidad de transmisión que el protocolo I²C. [11]

Este protocolo fue desarrollado por Motorola (ahora parte de NXP Semiconductors) aproximadamente en 1985. Es uno de los protocolos seriales síncronos más versátiles y más utilizados en el mundo de los microcontroladores, por su facilidad de implementación y su velocidad de comunicación en distancias cortas. Está basado en la arquitectura maestro-esclavo, en el que el dispositivo maestro tiene el control del bus, el cual pone los datos y genera la señal de reloj que se enviarán al dispositivo esclavo en particular.

La sincronización y la transmisión de datos se realiza por medio de 4 señales:

- SCLK (Clock): Es el pulso que marca la sincronización. Con cada pulso de este reloj, se lee o se envía un bit.
- MOSI (Master Output Slave Input): Salida de datos del Master y entrada de datos al Esclavo.
- MISO (Master Input Slave Output): Salida de datos del Esclavo y entrada al Master.
- SS/Select: Para seleccionar un Esclavo, o para que el Master le diga al Esclavo que se active.

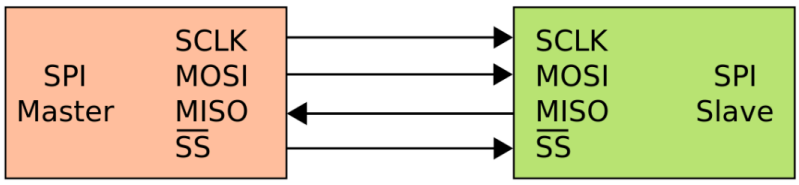


Imagen 19: Bus SPI

2.9 Soluciones comerciales existentes

Para desarrollar nuestro sistema es necesario conocer las soluciones comerciales que existen actualmente, ya que al analizar sus limitaciones, será posible diseñar un sistema personalizado, que permita mejorar tanto las funcionalidades, como el precio de los sistemas comerciales.

2.9.1 Programadores de riego

Los programadores de riego son dispositivos que permiten establecer una periodicidad en el riego, pudiendo estimar el agua requerida para un cultivo, la cantidad y momento adecuado, con el objetivo de maximizar la producción y obtener un producto de calidad y libre de enfermedades.
Generalmente, estos programadores de riego se encargan de regular la apertura y cierre de electroválvulas, los cuales permiten o no el paso de agua hacia las plantas.
En el mercado podemos encontrar programadores de riego sencillos, como por ejemplo los programadores de grifo, en los que el dispositivo se conecta a la boca de un grifo y mediante una placa de control se abre o cierra una pequeña electroválvula que tiene alojada en su interior. Estos están normalmente destinados para zonas pequeñas de riego, por lo que no serían útiles para cuando se necesiten grandes zonas de riego. Además, generalmente su instalación es en el exterior a la intemperie, por lo que se reduce considerablemente su vida útil. Por otro lado, existen programadores de riego más complejos, que permiten tener varias electroválvulas conectadas independientemente, por lo que es posible disponer de varias zonas de riego diferenciadas y con programaciones distintas en cada una.





Imagen 20 y 21: Programador de grifo y Programador de riego 4 zonas, programación flexible

2.9.2 Sistemas inteligentes de riego

Gracias a estos sistemas es posible acercarse más a la eficiencia del riego, ya que además de poder disponer de una o varias zonas de riego y poder programar uno o más riegos diarios con la cantidad y momento adecuado, se analizan las condiciones ambientales, tales como temperatura, humedad ambiental y humedad del suelo, para determinar si realmente es necesario realizar el riego o no. Esto mejora la eficacia de éste, ya que únicamente se regará cuando las condiciones sean las necesarias para ello, no aportando un exceso o defecto de agua.
Una de las dificultades que podríamos ver de estos sistemas es que son costosos en términos de instalación y dependiendo de la extensión de la zona de riego en donde se quiera implementar, influirá en su costo final [13].

CAPÍTULO III

DISEÑO DEL SISTEMA

ESPECIFICACIONES Y CARACTERÍSTICAS DEL SISTEMA
DISEÑO DE HARDWARE
ARQUITECTURA DEL SISTEMA
ESPECIFICACIONES Y CARACTERÍSTICAS DE LOS COMPONENTES
DISEÑO DE SOFTWARE
IMPLEMENTACIÓN DEL SISTEMA

3.1 Especificaciones y Características del Sistema

A continuación se detallarán los requerimientos y características necesarias para la funcionalidad del sistema.

Se pretende diseñar un prototipo de un sistema de riego automatizado, que según las condiciones meteorológicas y ambientales, sumado a las necesidades hídricas de los cultivos, permita gestionar de manera autónoma el suministro de agua. Con la finalidad de conseguir un riego óptimo, minimizando así, el consumo de agua.
Para poder llevar a cabo esta gestión autónoma del suministro de agua, se necesita recibir información de determinadas magnitudes, como la temperatura ambiental, la humedad ambiental, la humedad del suelo, la luminancia, etc. Dichos parámetros se registrarán mediante el uso de distintos sensores para posteriormente analizar los datos y determinar si es necesario realizar un aporte hídrico. A través de estos datos enviados por los sensores permitirán activar y desactivar los relés que hacen funcionar las bombas eléctricas de agua del sistema de riego.

Para lograr la eficiencia en el riego, éste se debe realizar en condiciones climáticas que permitan el mejor aprovechamiento posible del agua suministrada, para esto, en el prototipo, se establecerán condiciones de temperatura, humedad ambiental y luminosidad ideales para que el riego se produzca. Si no se cumplen dichas condiciones ambientales no se activará el riego, evitando así la evaporación del aporte hídrico sobrante.

Los datos suministrados por los sensores, serán registrados por el microcontrolador Arduino y podrán ser visualizados por el usuario, mediante una interfaz simple e intuitiva, como lo es una aplicación móvil.
Para la comunicación entre Arduino y la aplicación móvil, lo más conveniente sería utilizar una tecnología inalámbrica para evitar tener que realizar cableados. Por esto se utilizará un módulo WiFi, el cual es un elemento de bajo costo y gran fiabilidad. La comunicación entre Arduino y el módulo WiFi se realiza mediante puerto Serie-UART, gracias a la librería SoftwareSerial de Arduino.

3.2 Diseño de Hardware

A continuación se describirán los componentes electrónicos utilizados en el prototipo, especificando sus características y la interacción que existe entre ellos.

3.2.1 Arquitectura del Sistema

Para realizar las especificaciones del sistema indicadas en el apartado anterior, se necesita disponer de una serie de componentes, tales como: sensores, controladores, actuadores, etc.
En este apartado se desarrollará el diagrama de bloques de dichos componentes y de sus relaciones, para visualizar la estructura global del sistema a desarrollar.
Posteriormente se ampliará la descripción de los componentes empleados.

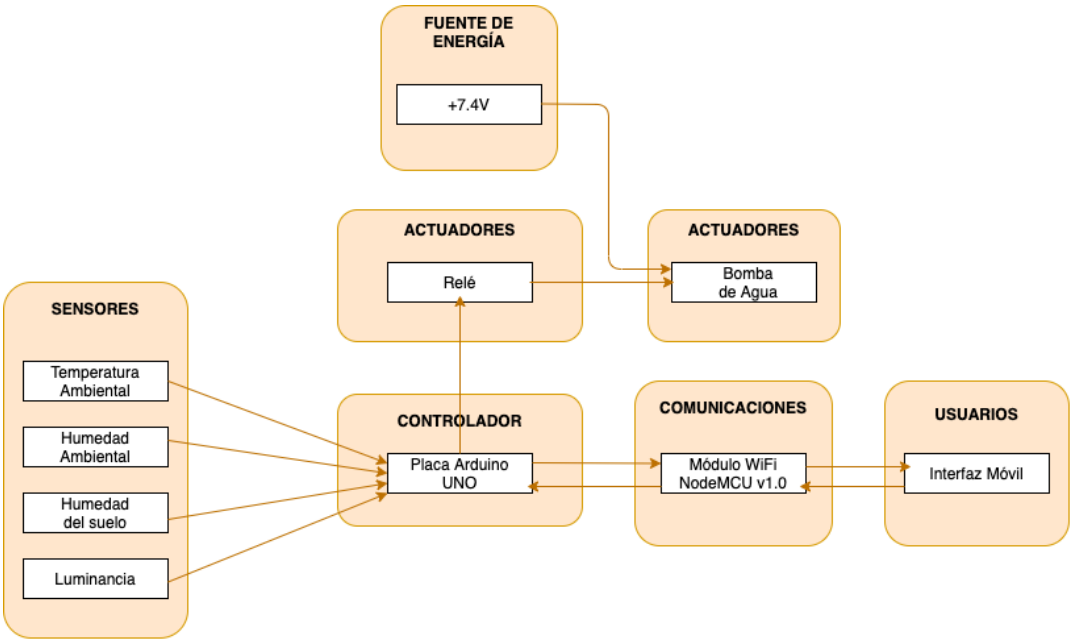


Imagen 22: Diagrama de bloques del sistema de riego automatizado

En el diagrama de bloques, podemos detallar los siguientes grupos [14]:

- **Sensores**: Se encuentran los sensores que se utilizarán para captar las condiciones climáticas y ambientales.

- **Controlador**: Es el elemento principal del sistema, el cual gestiona la información captada por los sensores y la analiza para enviar las acciones necesarias sobre los actuadores.
- **Actuadores**: Son la interfaz de salida por el cual se controla la realización del riego.
- **Comunicaciones**: Es el elemento mediante el cual los usuarios se comunican con el sistema, permitiendo una conexión inalámbrica.
- **Usuarios**: Son los encargados de interactuar con el sistema, los cuales mediante una aplicación móvil, podrán visualizar los datos y el estado del riego.

3.2.2 Especificaciones y Características de los componentes

3.2.2.1 Arduino Uno

Arduino Uno es una placa de microcontrolador open-source basado en el microchip ATmega328P y desarrollado por Arduino.cc. La placa está equipada con conjuntos de pines de entrada y salida, que pueden conectarse a varias placas de expansión y otros circuitos. La placa tiene 14 pines digitales, 6 pines analógicos y programables con el entorno de desarrollo de Arduino a través de un cable USB.
La placa puede ser alimentada por el cable USB o por una batería externa de 9 voltios, aunque también acepta voltajes entre 7 y 20 voltios.

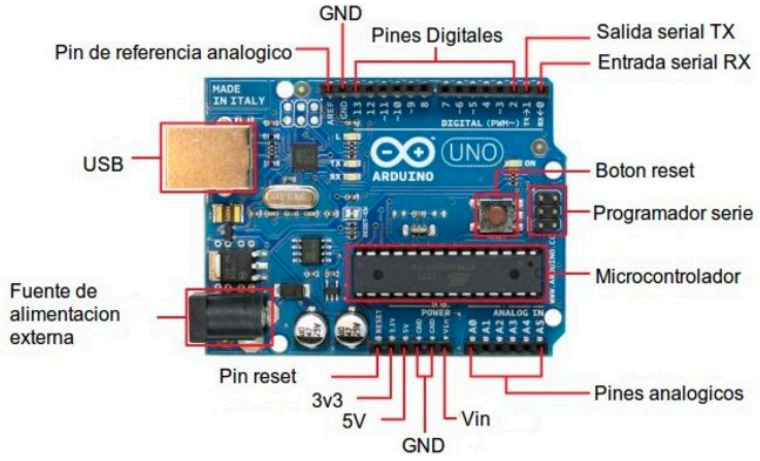


Imagen 23: Placa Arduino Uno R3

	Arduino Uno Rev 3
Microcontrolador	ATmega328P
Voltaje	5 V
Input Voltaje (Recomendado)	7-12 V
Input Voltaje (Límite)	6-20 V
Digital (I/O) Pins	14 (4 for PWM Output)
PWM Digital (I/O) Pins	6
Input Pins Analógicos	6
Canales Entrada analógica (multiplexed)	
DC Current por I/O Pin	20 mA
DC Current Por 3.3 V Pin	50 mA
Memoria Flash 32 KB (.5 KB for bootloader)	32 KB (4 KB for bootloader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz
LED BUILTIN	Pin 13
Bluetooth	no
Longitud	68.6 mm
Anchura	53.4 mm
Peso	25 g

Tabla N° 3: Características de placa Arduino Uno R3

En la siguiente imagen podemos visualizar la disposición y la comunicación de los pines de Arduino Uno R3.

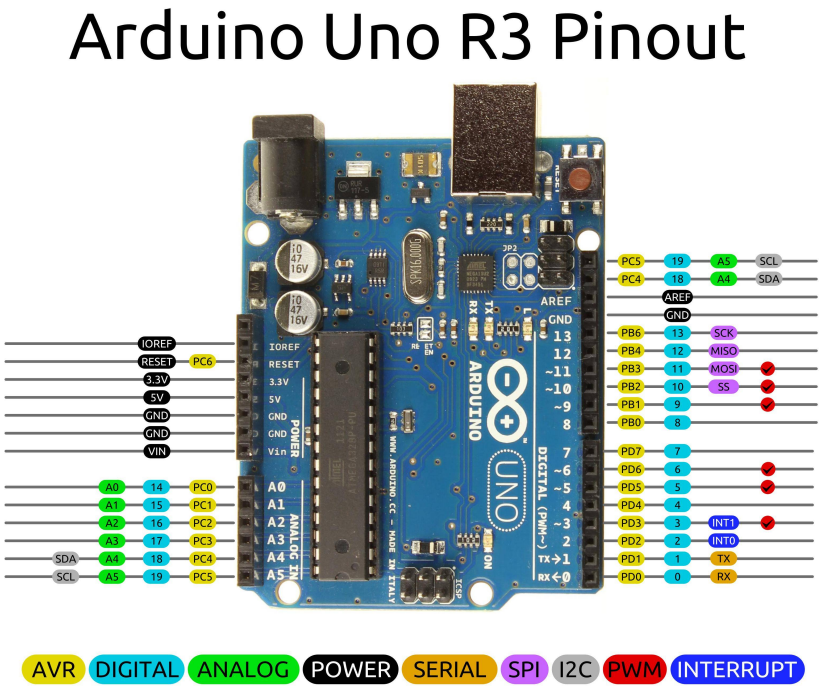


Imagen 24: Pines placa Arduino Uno R3

3.2.2.2 Sensor de Temperatura y Humedad del aire - DHT22

El DHT22 es un sensor de temperatura y humedad relativa del aire de bajo costo. 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. Es bastante simple de usar, pero requiere sincronización cuidadosa para tomar datos.
El DHT22 es de dimensiones pequeñas con bajo consumo de energía y con una muy buena distancia de transmisión de datos de hasta 20 metros. 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 mínimo cada 2 segundos.

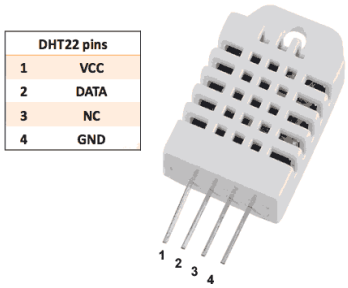


Imagen 25: DHT22, Sensor de temperatura y humedad ambiental

Model	DHT22
Power supply	3.3-6V DC
Output signal	digital signal via single-bus
Sensing element	Polymer capacitor
Operating range	humidity 0-100%RH; temperature -40~80Celsius
Accuracy	humidity +-2%RH(Max +-5%RH); temperature <+-0.5Celsius
Resolution or sensitivity	humidity 0.1%RH; temperature 0.1Celsius
Repeatability	humidity +-1%RH; temperature +-0.2Celsius
Humidity hysteresis	+-0.3%RH
Long-term Stability	+-0.5%RH/year
Sensing period	Average: 2s
Interchangeability	fully interchangeable
Dimensions	small size 14*18*5.5mm; big size 22*28*5mm

Tabla N° 4: Características Generales sensor DHT22

3.2.2.3 Sensor de Humedad del suelo - Sensor Capacitivo de humedad v1.2

El Sensor de humedad de suelo Capacitivo v1.2 permite medir la humedad en el suelo utilizando el principio de capacitancia entre electrodos en lugar de resistencia como otros tipos de sensores de humedad, lo que aumenta considerablemente la vida útil del sensor. El electrodo posee una capa de protección anticorrosión para una mayor duración.
El funcionamiento del sensor se basa en medir la capacitancia entre 2 electrodos insertados dentro del suelo, la capacitancia entre los electrodos dependerá de la humedad del suelo, por lo que para un suelo muy húmedo tendremos una capacitancia muy baja y para un suelo muy seco la capacitancia será muy alta. El electrodo va conectado a una tarjeta de acondicionamiento que entrega una salida analógica. La salida analógica (AO) entrega un voltaje analógico desde 0V para un suelo muy húmedo hasta 5V para un suelo muy seco.
Este sensor incluye un regulador de tensión que provee la capacidad de ser energizado en un rango de voltaje operativo de 3.3 a 5V, siendo así compatible con gran cantidad de microcontroladores y tarjetas de desarrollo como Arduino.

Voltaje de alimentación: 3.3V - 5V DC
Corriente operación: 5mA
Voltaje de la señal de salida: 0 a 5V (Analógico)
Conector: PH2.0-3P
Dimensiones: 98*23 mm
Peso: 15 gramos



Imagen 26: Sensor Capacitivo de humedad de suelo v1.2

3.2.2.4 Sensor de luminosidad BH1750

EL módulo BH1750 es un sensor digital de nivel de luz que puede ser conectado con facilidad a un procesador como Arduino y nos entrega valores de medición en Lux (lumen/m²) que es una unidad de medida estándar para el nivel de iluminación (iluminancia[10]).
La comunicación entre este sensor y Arduino se realiza mediante el protocolo I²C, por lo que es sencillo obtener los datos medidos.
El sensor BH1750 tiene un amplio rango de medición ajustable ente 0.11 – 100.000 lux y una alta precisión, por lo que podrá determinar la iluminación existente en casi cualquier situación.
A continuación se presentan algunos ejemplos de iluminancia (Lux) para distintas situaciones [15]:

Situación	Lux
Noche	0.001-0.02
Luna Llena	0.2-0.6
Día nublado, en interior	5-50
Día nublado, en exterior	50-500
Día soleado, en interior	100-1000
Bajo luz directa del sol	Alrededor de 100.000

Tabla N° 5: Ejemplos de iluminancia

Sensor con BH1750 - Modelo GY-302
Voltaje de alimentación: 3.3 – 5V
Salida: Digital
Rango de luz: 0-65535 lx
Número de pines: 5 (Vcc, GND, SCL, SDA, ADDR)
Dimensiones: 13,9 x 18,5mm

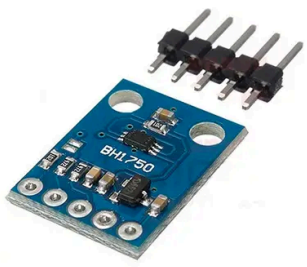


Imagen 27: Sensor BH1750

[10] iluminancia: Es la relación entre el flujo luminoso (la cantidad de luz emitida por una fuente de luz), y la superficie en la que se mide.

3.2.2.5 Módulo Relé (relay) de 4 Canales

Mediante el conjunto de relés se controlarán las bombas de agua necesarias para la realización del riego.
Este módulo de relés nos permitirá controlar componentes de alto voltaje o alto amperaje, como lo son las bombas de agua, las que no podrían ser manejadas directamente con Arduino. Por lo tanto es necesario utilizar Relays o Relés, ya que estos dispositivos permiten controlar cargas de alto voltaje con una señal pequeña.
Este módulo Relay activa la salida normalmente abierta (NO: Normally Open) al recibir un "0" lógico (0 Voltios) y desactiva la salida con un "1" lógico (5 voltios).

Modelo: JQC-3FF-S-Z
Voltaje de Alimentación: 5V
Canales: 4
Carga máxima de salida: 10A/250VAC
Corriente máx: 10A (NO), 5A (NC).
Dimensiones: 75 mm x 55 mm x 19.3 mm

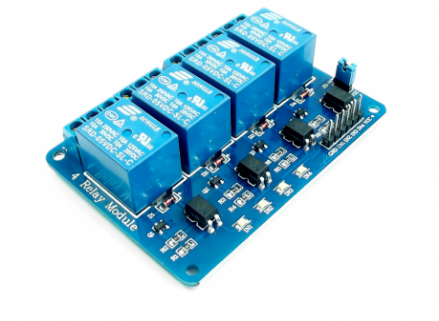


Imagen 28: Módulo Relé de 4 canales

3.2.2.6 Bomba de agua sumergible

Esta mini-bomba de agua sumergible, es una bomba diseñada para poder usar en proyectos pequeños como fuentes de agua, sistemas de riego automático, hidroponía, etc. A diferencia de otras bombas de agua, ésta tiene la capacidad de mover líquidos desde un punto a otro, estando dentro del recipiente en donde se desea extraer el contenido. Puede mover líquidos desde un estanque, un recipiente o cualquier fuente de agua hasta cualquier otro punto que se encuentre a una altura no mayor a 40cms de esta bomba.
Es compatible con Arduino, pero es necesario el uso de un Relé o Transistor capaz de manejar su consumo.



Imagen 29: Bomba de agua sumergible

3.2.2.7 Node MCU

Para la realización de las comunicaciones del sistema se utilizará la placa ESP8266 Node MCU v1.0, la cual dispone de WIFI por lo que podrá conectarse a Arduino y enviar las lecturas de todos los datos que entreguen los sensores a la nube.
La comunicación entre la placa ESP8266 y Arduino se realiza a través de puerto serie. Mediante la librería SoftwareSerial.h, podremos utilizar los pines D1 (RX) y D2 (TX) para realizar dicha conexión serie. En la siguiente imagen podemos visualizar la disposición de estos pines.

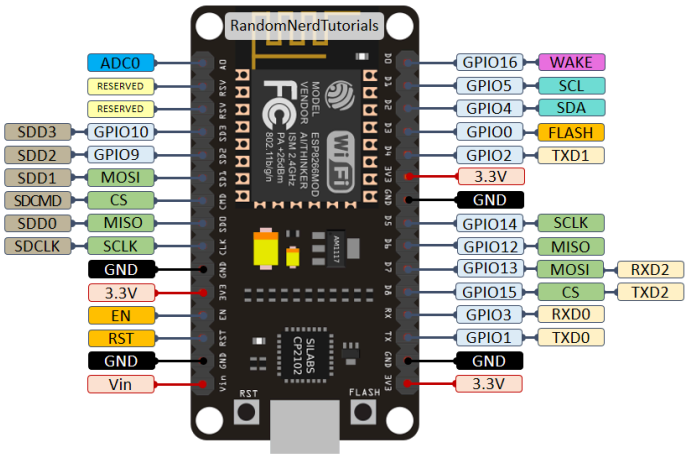


Imagen 30: Pines placa ESP8266 Node MCU v1.0

Modelo: ESP8266 ESP-12E v1.0 / V2
Voltaje de entrada: 3.3V
Pines digitales de salida: 13
Wi-Fi 2.4 GHz, soporta WPA/WPA2
Protocolo Integrado: TCP/IP

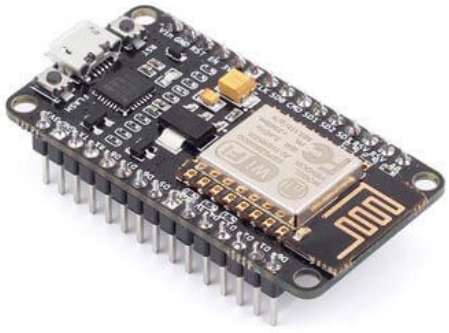


Imagen 31: Placa ESP8266 Node MCU v1.0

3.2.2.8 Display LCD 20x4

Esta pantalla LCD 20x4 permite visualizar mensajes y caracteres que provengan de cualquier microcontrolador, especialmente Arduino.
Puede desplegar 20 caracteres en cuatro filas, completando un total de 80 caracteres alfanuméricos, e incluso diferentes símbolos que ayudarán a potenciar la información que se va a mostrar en él. Es compatible con muchas librerías existentes en la web.

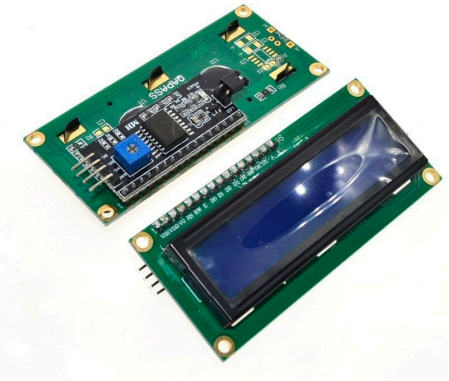


Imagen 32: Display LCD 20x4

3.3 Diseño de Software

3.3.1 Entorno de Desarrollo

Existen distintos entornos de desarrollo, pero para la implementación del proyecto se ha decidido hacer uso del IDE de Arduino, ya que es más sencillo de utilizar, tiene mayor flexibilidad y comodidad en el uso de librerías.

Un programa diseñado para Arduino sigue una estructura básica que siempre se compone de tres secciones [16]:

- Declaración de variables globales, funciones y librerías.
- La sección llamada **"void setup()"**: Se ejecuta una única vez, en el momento de encender (o reiniciar) la placa Arduino y suele ser utilizado para realizar las configuraciones iniciales, es decir, establecer cuales serán los pines de entrada o salida, configurar parámetros como velocidad o frecuencias, y todos los demás acuerdos necesarios para poder establecer todas las compatibilidades con nuestro hardware complementario.
- La sección llamada **"void loop()"**: Se ejecuta inmediatamente después de la sección "void setup()" de manera infinita y cíclica, hasta que se interrumpe la alimentación de la placa o se reinicia.

Las funciones, nos permiten tener un programa organizado, ya que encapsulan diferentes acciones que se deseen realizar de forma repetitiva. Estas funciones son llamadas en cualquiera de las tres secciones anteriores.
Las variables nos permiten almacenar valores para su uso posterior, asignándoles un nombre. Este valor puede ser modificado en cualquier momento por nuestro programa.

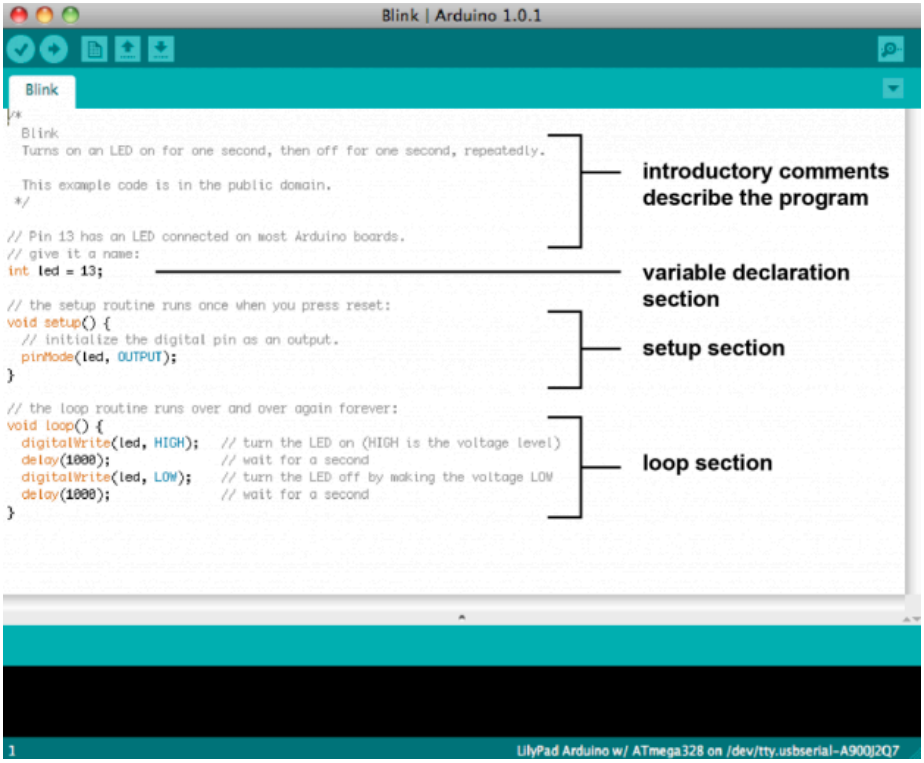


Imagen 33: Entorno de desarrollo IDE Arduino

3.3.2 Algoritmo

El sistema se realizó mediante un algoritmo estructurado con la siguiente secuencia:

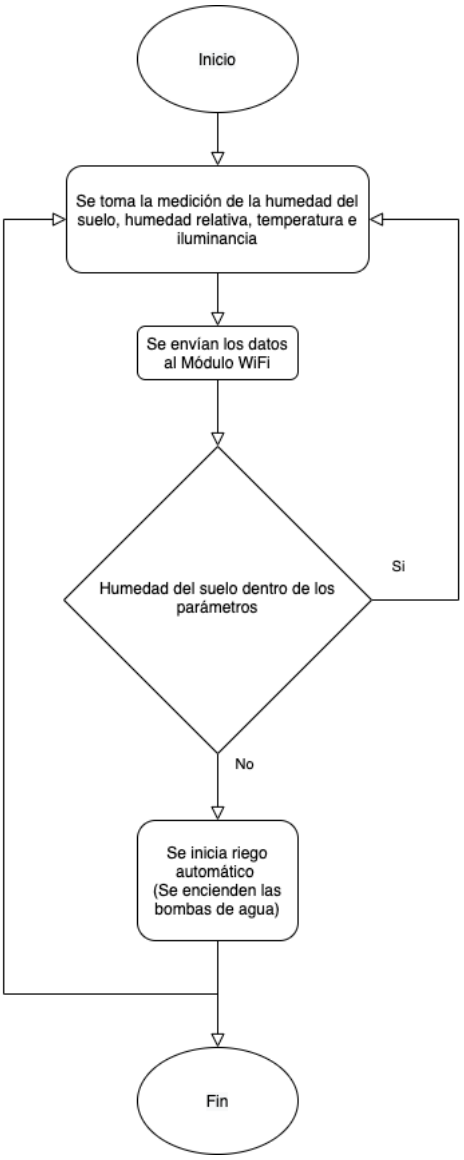
Inicio

- Se inician las mediciones con el sensor de humedad del suelo.
- Se clasifica la medición tomada por el sensor de humedad del suelo
    - De 579 a 526 → suelo seco
    - De 526 a 250 → suelo húmedo
    - De 250 a 220 → suelo encharcado.
- Se toma lectura de la humedad relativa y temperatura con el sensor DHT22.
- Se toma lectura de la iluminancia con el sensor BH1750.
- Si el valor de la humedad de suelo es menor a 526, se continua monitoreando la humedad del suelo. En caso contrario, se activa la bomba de riego.
- Si se activa la bomba de riego, al ser desactivada después de un minuto de riego se continua el monitoreo.
- Se envían todos los datos constantemente al módulo WiFi NodeMCU.
- El módulo WiFi NodeMCU envía constantemente los datos recibidos a una base de datos en tiempo real (Firebase).
- Los datos almacenados en la base de datos, son utilizados por una aplicación móvil y mostrados al usuario en una interfaz intuitiva.

Fin

Para el sistema se dispondrán de dos programas diferentes en nuestro IDE (sketch), uno para el controlador Arduino, que contiene las conexiones a los sensores y otro para el módulo WiFi, para el envío de los datos a la base de datos de Firebase.

A continuación se presentan los diagramas de flujos para cada uno de estos módulos.



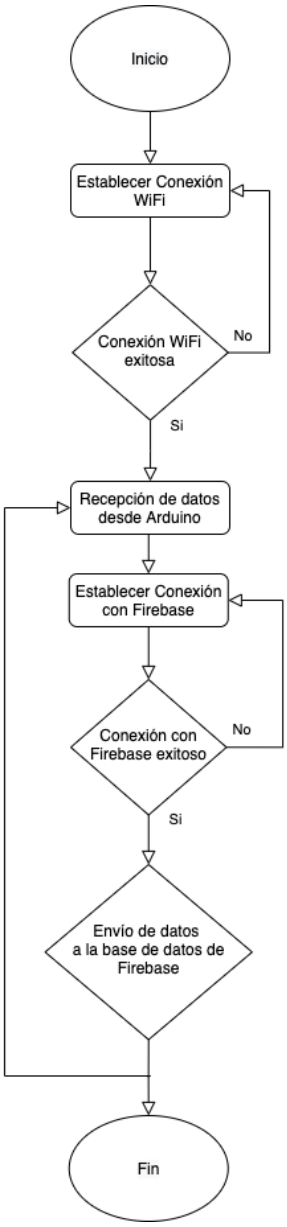


Imagen 34 y 35: Diagrama de flujo Placa Arduino y Diagrama de flujo Módulo WiFi NodeMCU

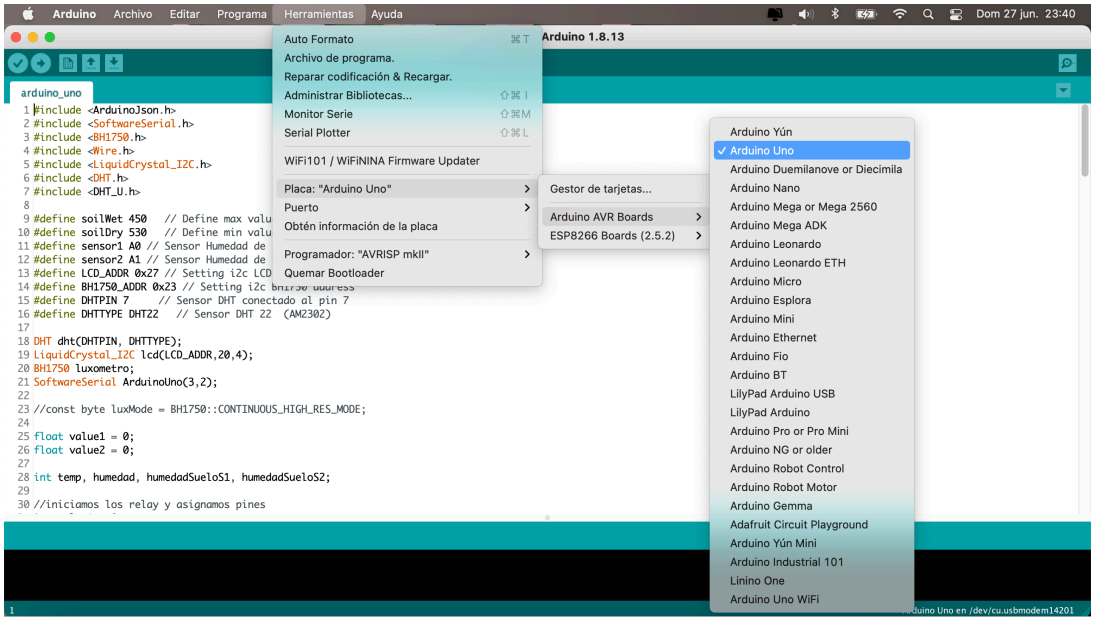


Imagen 36: Configuración Arduino Uno en IDE de Arduino

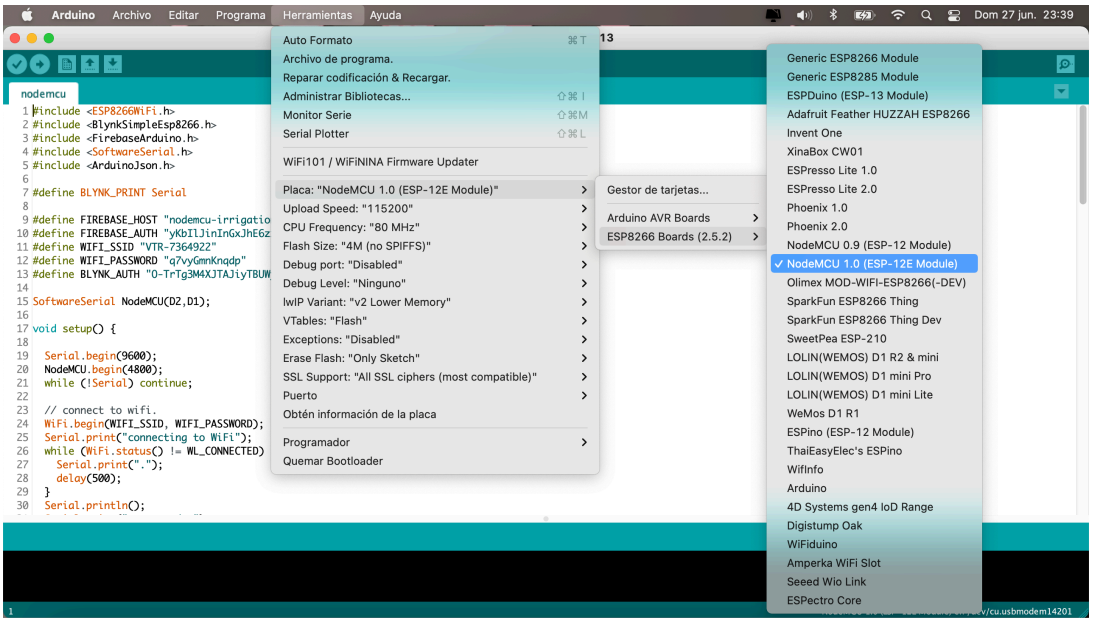
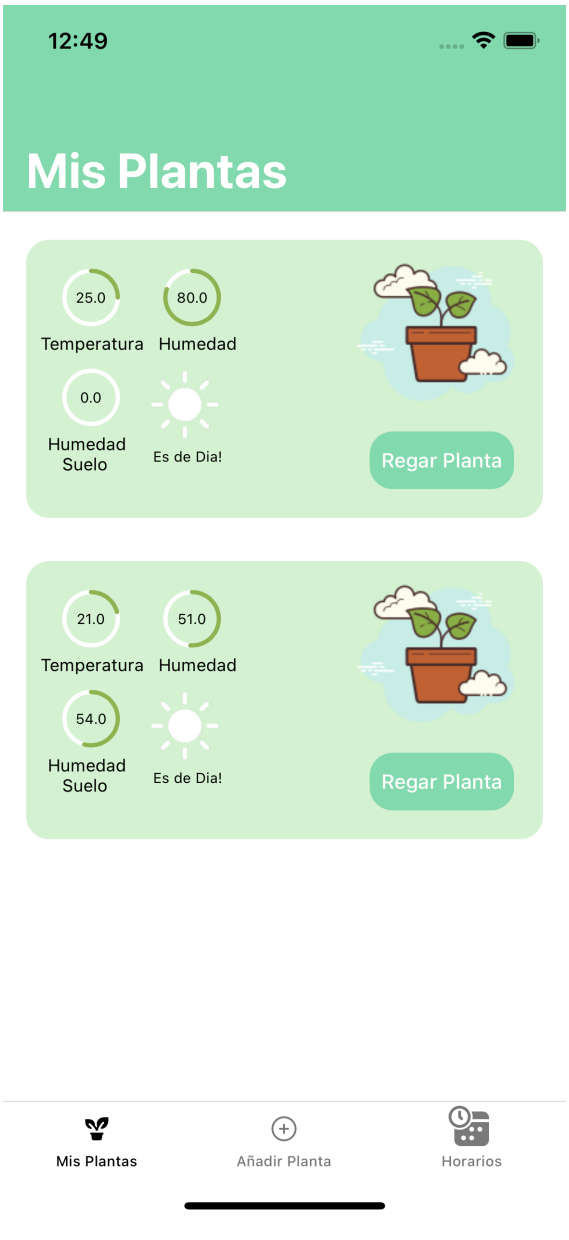


Imagen 37: Configuración módulo WiFi NodeMCU v1.0 en IDE de Arduino

3.3.3 Aplicación Móvil

Para poder visualizar los datos que nos entreguen los sensores de manera más intuitiva para el usuario, se decidió diseñar e implementar una aplicación móvil en iOS que le permita al usuario tener el control del riego y poder monitorizar sus plantas con los datos obtenidos en tiempo real.
En una primera instancia, la aplicación mostrará en la sección “Mis Plantas” el listado de plantas que tenga asociado el usuario en el sistema, es decir, si se tiene un sensor de humedad de suelo, se podrá monitorear a una planta. Si se tienen más sensores de humedad de suelo se mostrarán más plantas en el listado, como por ejemplo la imagen 38.
En esta sección de “Mis Plantas” se mostrarán los valores obtenidos de la temperatura, la humedad, la humedad del suelo y la iluminancia. Si el nivel de iluminancia determina que es de día se mostrará un mensaje “Es de dia” y si el sensor determina que es de noche, se mostrará un mensaje de “Es de noche”, como la imagen 39. Con esto el usuario tendrá la información necesaria para tomar la decisión de regar a ciertas horas del día.
Además en esta sección, se podrá realizar un regado de tipo “Manual”. Para realizar este regado se puede presionar el botón verde de “Regar Planta” y también se puede desactivar el riego, presionando el botón rojo de “Desactivar Riego”, como la imagen 40. Se hizo una diferencia en el color del botón siguiendo el diseño de UX/UI para una mejor experiencia en el usuario.
A futuro, lo ideal sería seguir trabajando en esta aplicación en la siguientes secciones, como por ejemplo, poder añadir una nueva planta, asociando su correspondiente sensor de humedad de suelo y pudiendo personalizar el nombre o tipo de planta, también sería ideal determinar horarios de riego y que la aplicación móvil pueda notificar al usuario cuando este horario se debe llevar a cabo.



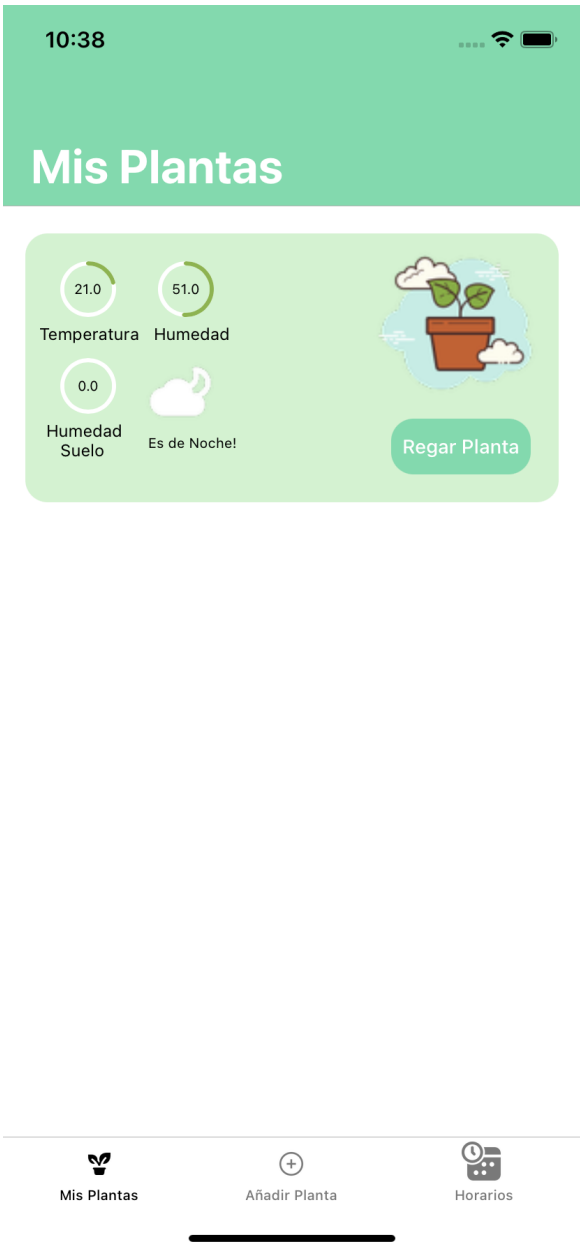


Imagen 38 y 39: Aplicación Móvil: Listado de Plantas y Nivel de Iluminancia “Noche"

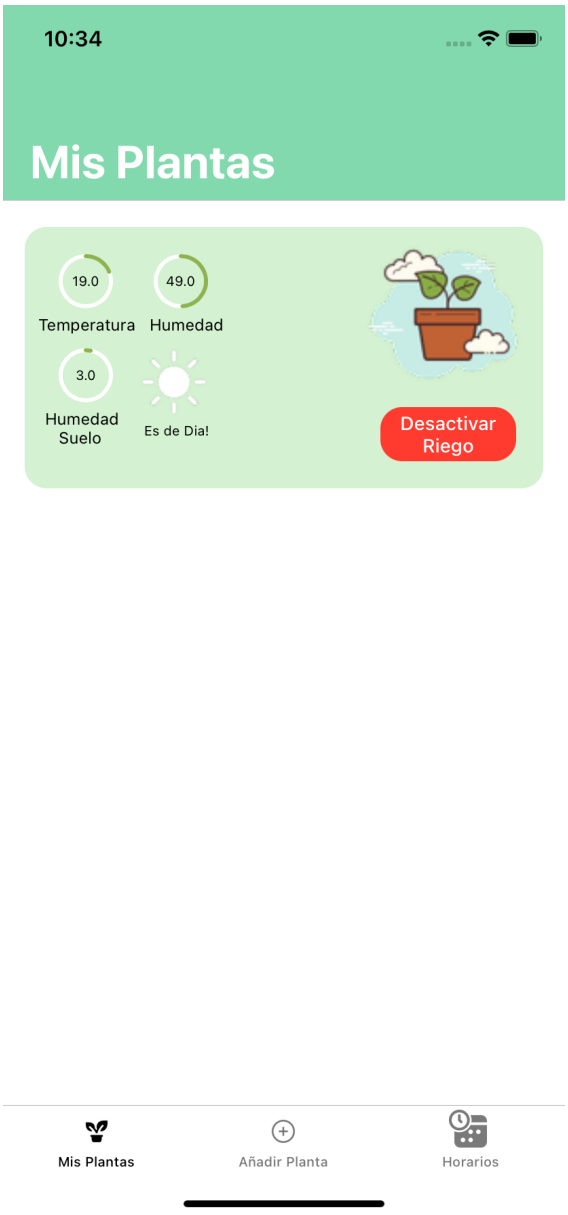


Imagen 40: Aplicación Móvil: Desactivar Riego

3.3.4 Librerías utilizadas

En esta sección se describirán las librerías que se han utilizado en los programas de Arduino y módulo WiFi, para poder añadir nuevas funcionalidades y gestionar los distintos componentes.
Hay tres diversos tipos de librerías de Arduino: base, estándar, y contribuciones.

- Librerías Base (Core Library): La biblioteca de la base forma parte del entorno de desarrollo del Arduino (IDE). El propósito de esta librería es ocultar mucha de la complejidad que tradicionalmente implica el trabajar con el microprocesador.
  La mayoría de los proyectos leen datos de una de las entradas o escriben datos en una de las salidas. La librería core hace que estas tareas comunes sean simples de ejecutar. Por ejemplo, para leer el valor de un pin digital basta con usar la función *digitalRead* o para escribir el valor de un pin digital basta con usar la función *digitalWrite.*
  En nuestro programa utilizaremos algunas funciones de base (core).

- Librerías Estándar: Estas librerías son desarrolladas por el equipo de Arduino, por lo tanto vienen preinstaladas en el entorno de desarrollo de Arduino (IDE).

  - **Librería Wire:** Se utiliza para comunicar la placa Arduino con dispositivos que trabajan mediante el protocolo I^2C. Los pines asignados para esta comunicación son SDA y SCL.
  Se incluye mediante el comando #include <Wire.h>.
  - **Librería SoftwareSerial:** Permite la comunicación serie en otro pines digitales de Arduino.
  Se incluye mediante el comando #include <SoftwareSerial.h>.

- Librerías por contribuciones: Estas librerías son contribuciones de los usuarios de Arduino que no se distribuyen como parte del entorno de desarrollo IDE. Se puede encontrar muchas de estas librerías listadas en el sitio web Arduino. Algunas de estas librerías son extensiones de las librerías estándar que añaden algunas funciones suplementarias, y con el tiempo, si estas funciones resultan ser útiles, el equipo de desarrollo puede añadirlas a las librerías estándar o incluso a la librería de base.

  - **Librería ArduinoJson:** Se encarga de la gestión de los datos JSON. Permite la serialización y deserialización de objetos y datos JSON.
  Para utilizar esta librería, se debe instalar mediante el gestor de librerías incluido en el IDE de Arduino. Para nuestro prototipo utilizaremos la versión 5.13.5 por problema de

compatibilidades. [17]
Se incluye mediante el comando #include <ArduinoJson.h>.
- **Librería BH1750:** Esta librería fue desarrollada por Christopher Laws y se utilizará para poder realizar las lecturas de iluminancia.
Para utilizar esta librería, se debe instalar mediante el gestor de librerías incluido en el IDE de Arduino o también descargarlo desde el siguiente enlace:
https://github.com/claws/BH1750.
Se incluye mediante el comando #include <BH1750.h>.
- **Librería LiquidCrystal_I2C:** Con esta librería podremos controlar nuestro display LCD 20x4 de manera muy simple y sencillo.
Para utilizar esta librería, se debe descargar desde el siguiente enlace y seguir los pasos para la instalación:
http://elexhub.com/wp-content/uploads/2019/06/ELEXHUB-Tutorial_4x20-Liquid-Crystal-Display-with-i2c.pdf.
- **Librería DHT:** Con esta librería podemos obtener los valores de temperatura y humedad de los sensores DHT, permite un fácil manejo de instrucciones para controlar nuestro sensor.
Para utilizar esta librería, se debe instalar mediante el gestor de librerías incluido en el IDE de Arduino o también descargarlo desde el siguiente enlace:
https://github.com/adafruit/DHT-sensor-library.
Se incluye mediante los comandos #include <DHT.h> y #include <DHT_U.h>.
- **Librería ESP8266WiFi:** Es la encargada de la conexión y configuración del WiFi.
Para utilizar esta librería, se debe seguir los pasos del siguiente enlace para integrar este módulo en el IDE de Arduino:
https://github.com/esp8266/Arduino.
Se incluye mediante el comando #include <ESP8266WiFi.h>.
- **Librería FirebaseArduino:** Esta librería permite simplificar la conexión a la base de datos de Firebase desde clientes Arduino. [18]
Para utilizar esta librería, se debe seguir los pasos del siguiente enlace para la correcta instalación:
https://github.com/mobizt/Firebase-ESP8266.
Se incluye mediante el comando #include <FirebaseArduino.h>.

3.4 Implementación del Sistema

3.4.1 Interconexiones

A continuación veremos el detalle de las conexiones realizadas para cada componente utilizado en el sistema, de esta forma podremos visualizar de mejor manera las conexiones de cableado de cada sensor de forma independiente.
Todos los diseños de los esquemas electrónicos fueron realizados con el programa Fritzing [19], el cual es un software open-source muy útil para la automatización de los procesos de diseño electrónico. Es posible crear desde los esquemáticos de circuitos, montar los prototipos en una placa de pruebas virtual hasta escoger el mejor ruteado del circuito para construir la PCB.

3.4.1.1 Sistema Completo

En este apartado se muestra el diagrama de conexión del circuito electrónico general del sistema.

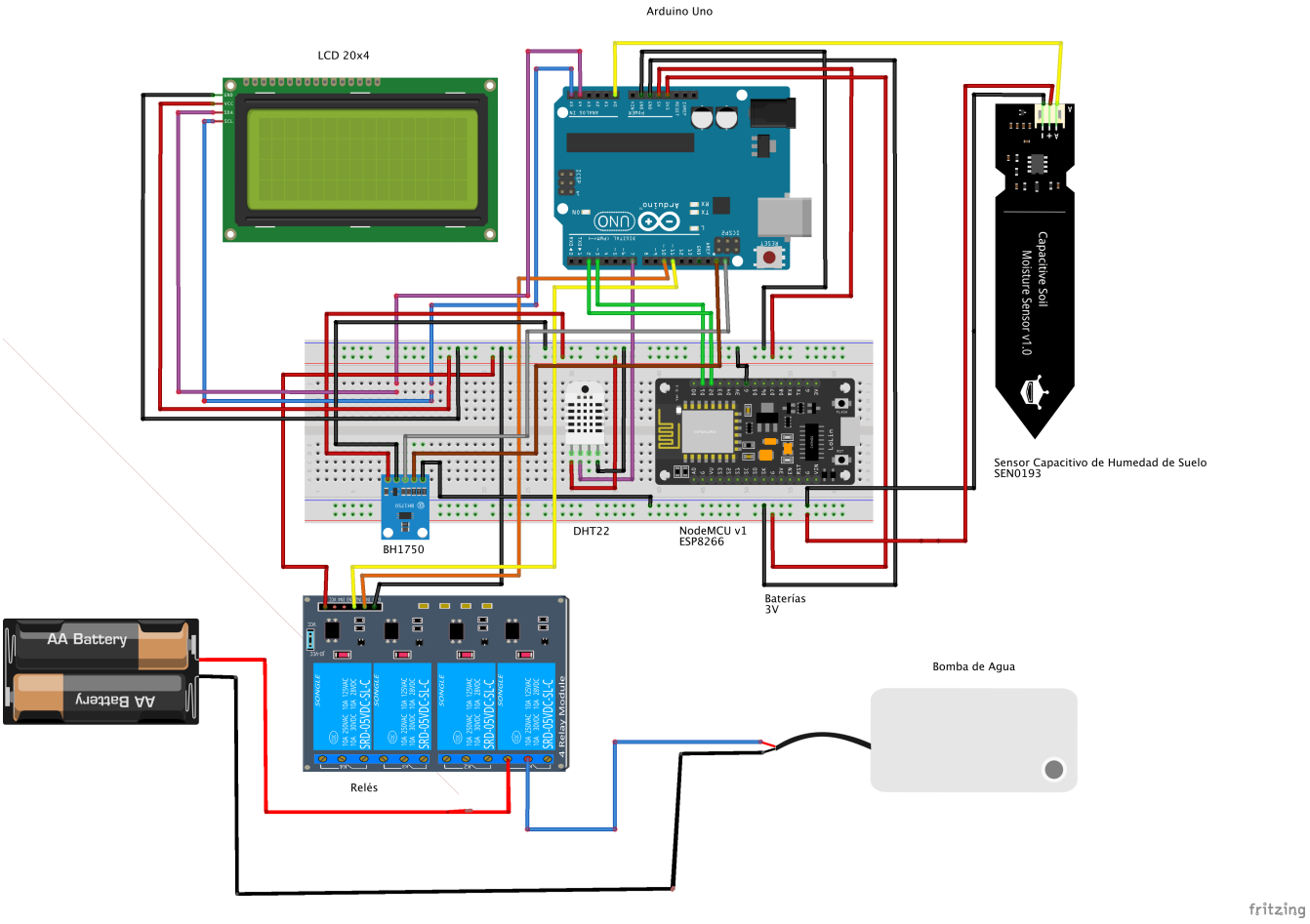


Imagen 41: Conexión Sistema Completo

3.4.1.2 DHT22

A continuación se muestran las conexiones realizadas entre Arduino y el sensor de temperatura y humedad ambiental DHT22.
Para este sensor utilizaremos una librería, por lo que podemos realizar fácilmente la lectura del sensor y no preocuparnos por el protocolo de comunicación entre Arduino y el sensor.
Por lo tanto las conexiones del sensor para nuestro prototipo son: El pin de dato a un pin digital en nuestro Arduino, en nuestro caso usaremos el pin 7 y la conexión de los pines de VCC y GND a los pines correspondientes de Arduino para el suministro de alimentación eléctrica.

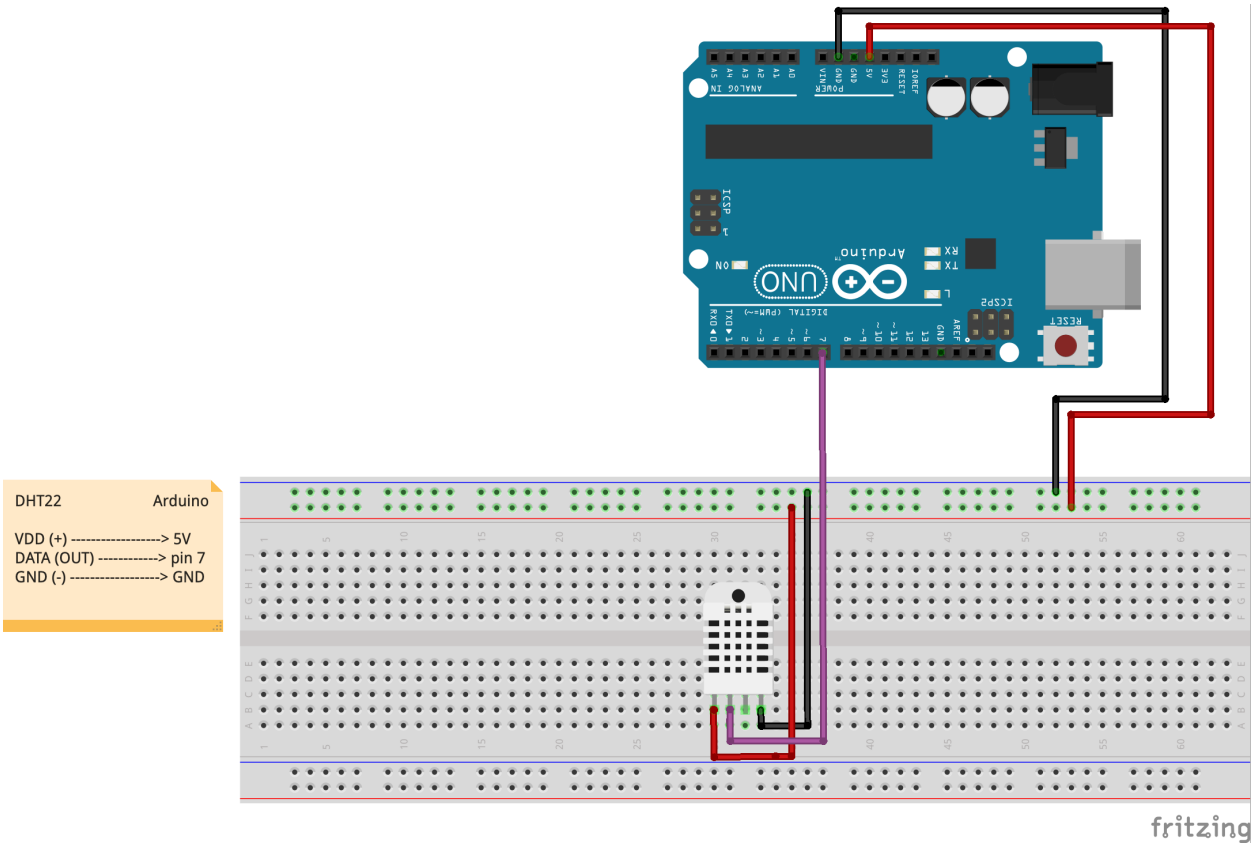


Imagen 42: Conexión Arduino y Sensor DHT22

3.4.1.3 Sensor Capacitivo de Humedad de Suelo

El sensor Capacitivo de Humedad de Suelo v1.2 es un sensor anticorrosivo encargado de medir la humedad en suelos. Este sensor produce una tensión en el pin A, inversamente proporcional al valor de humedad presente en la tierra.
Para nuestro prototipo se utilizará un sensor, el cual nos permitirá medir la humedad del suelo de una planta. En el caso de querer hacer mediciones a más plantas, se deberán agregar más sensores.
Por lo tanto para nuestro prototipo se realizan las siguientes conexiones: A0 de Arduino al pin A del sensor. Además de la conexión de los pines de VCC (+) y GND (-) a los pines correspondientes de Arduino para el suministro de alimentación eléctrica.

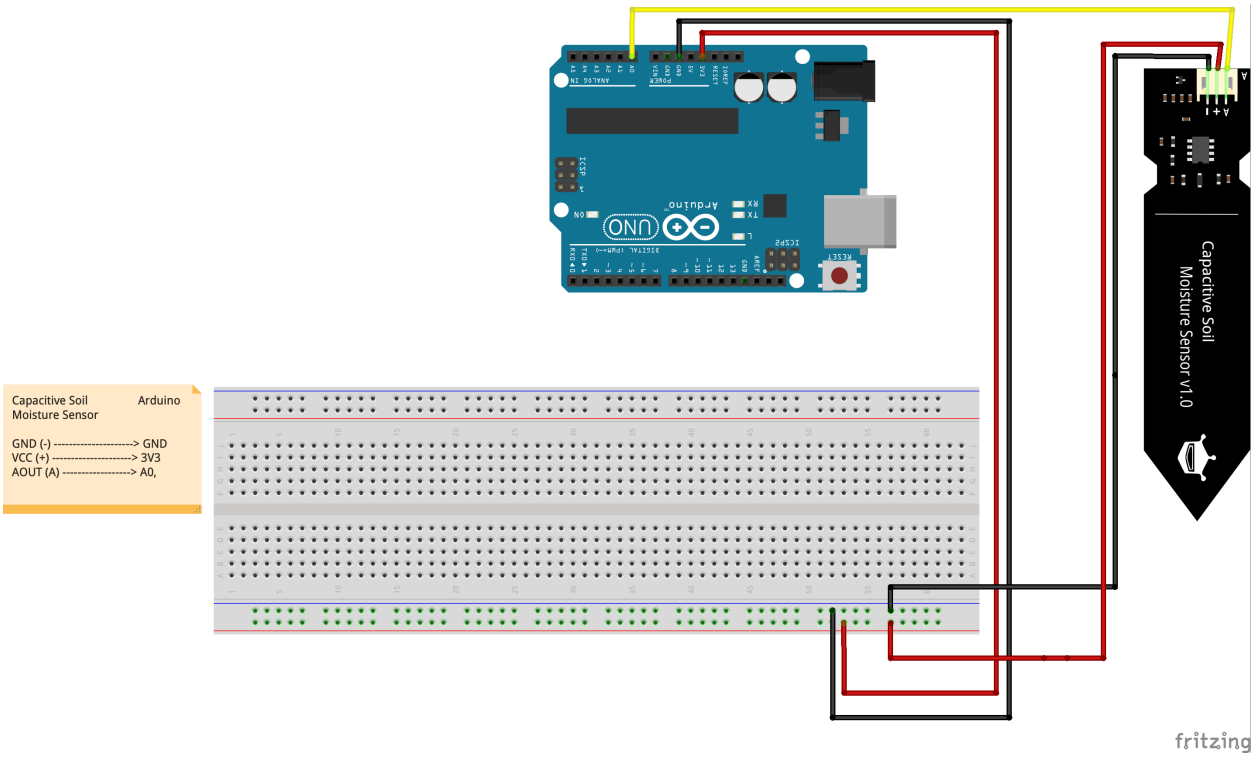


Imagen 43: Conexión Arduino y Sensor Capacitivo de Humedad de Suelo v1.2

3.4.1.4 Sensor de luminosidad BH1750

El sensor BH1750 está encargado de la realización de las mediciones de iluminancia. La comunicación de datos entre este sensor y Arduino se realiza mediante protocolo I²C, por lo que para la conexión del mismo realizaremos el cableado entre los siguientes pines:
SDA con pin SDA de Arduino, SCL con pin SCL de Arduino y los pines de VCC y GND a los pines correspondientes de Arduino para el suministro de alimentación eléctrica.

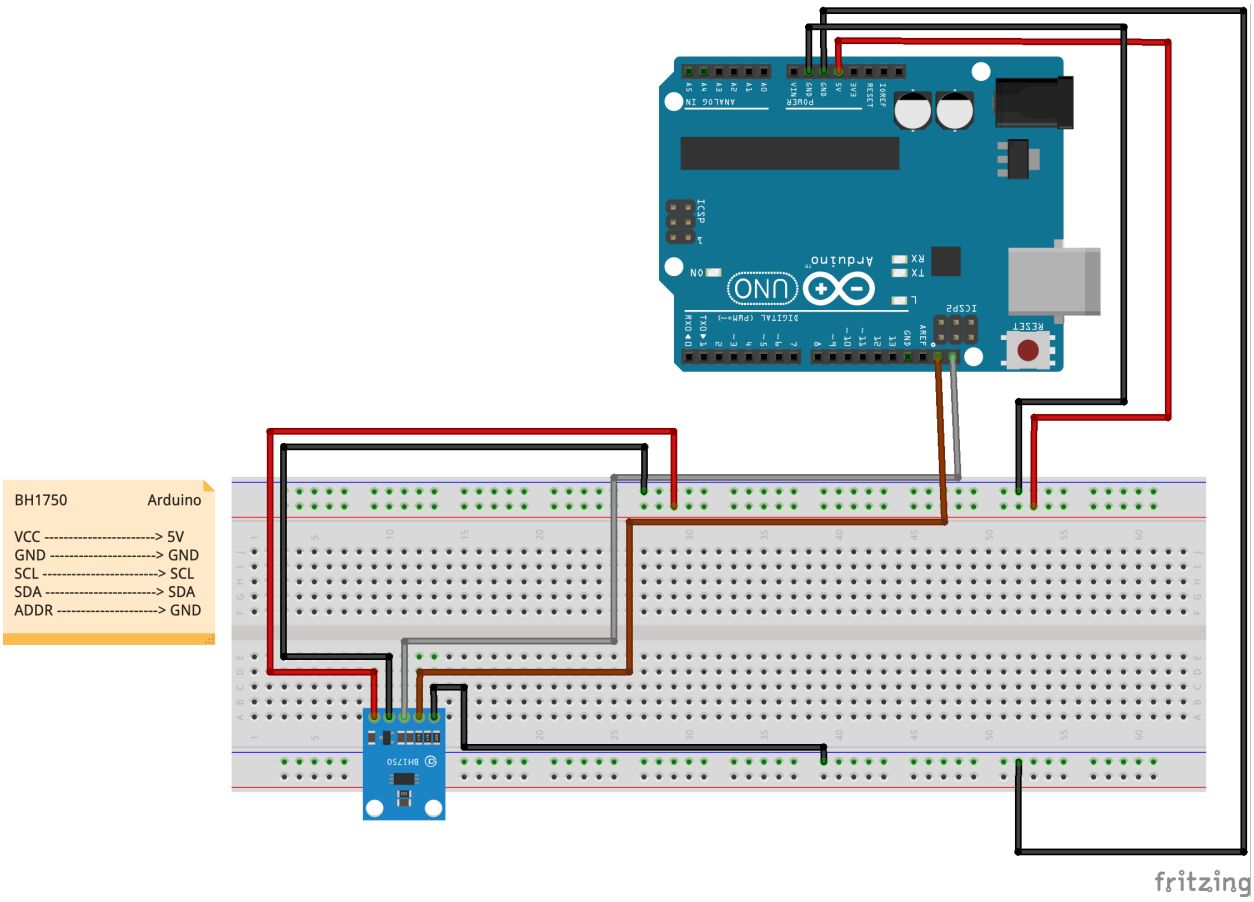


Imagen 44: Conexión Arduino y Sensor BH1750

3.4.1.5 Dispositivo LCD

El dispositivo LCD nos permite visualizar de manera más intuitiva todos los datos que nos entreguen los sensores.
La comunicación de datos entre el dispositivo LCD y Arduino se realiza mediante protocolo I²C, por lo que para la conexión del mismo realizaremos el cableado entre los siguientes pines:
SDA con pin A4 de Arduino (reservado para SDA), SCL con pin A5 de Arduino (reservado para SCL) y los pines de VCC y GND a los pines correspondientes de Arduino para el suministro de alimentación eléctrica.

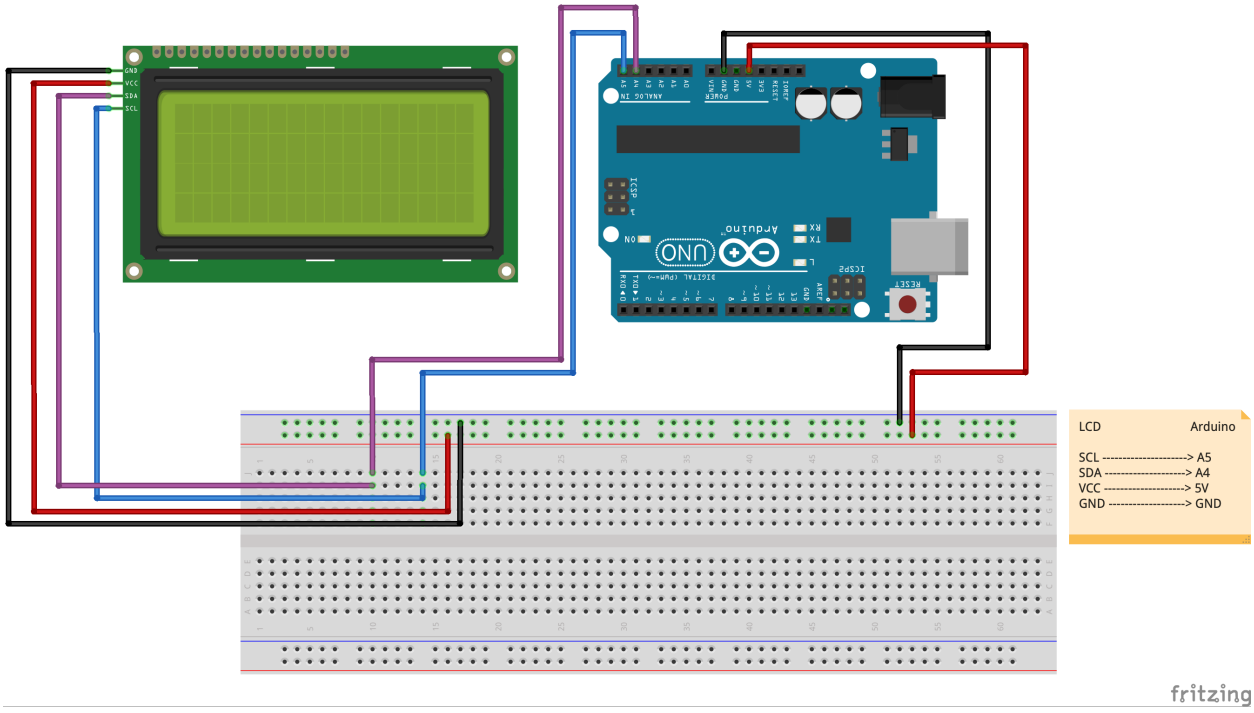


Imagen 45: Conexión Arduino y dispositivo LCD

3.4.1.6 Node MCU

A continuación se muestran las conexiones realizadas entre Arduino y el módulo WiFi NodeMCU v1.0. Este dispositivo dispone de una conexión a tierra (GND) común con Arduino y la comunicación entre el microcontrolador y el módulo NodeMCU se realiza en serie, para esto se utiliza los pines digitales D1 y D2.

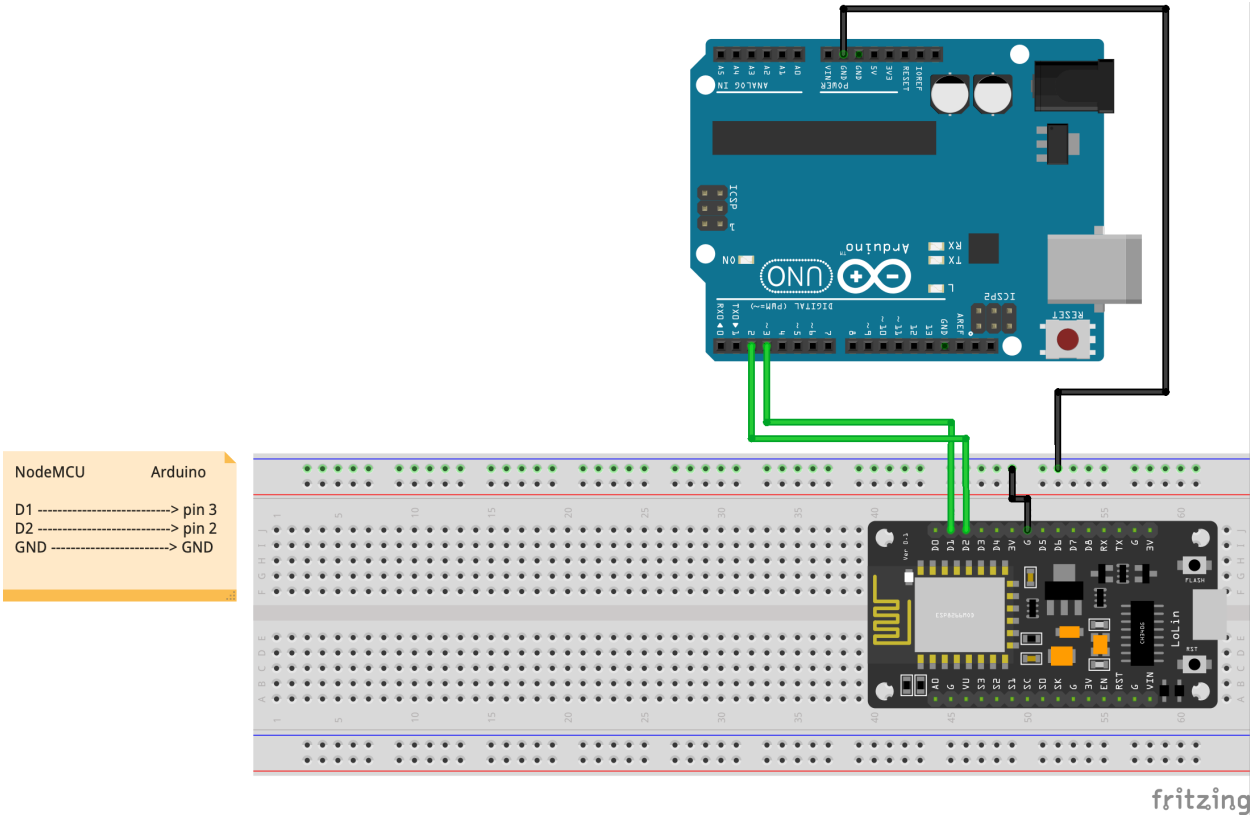


Imagen 46: Conexión Arduino y módulo WiFi NodeMCU v1.0

3.4.1.7 Relé

Los relés dispones de un pin de activación, el cual funciona con lógica inversa, es decir, excita la bobina y por tanto consideraremos que está activado el relé con un nivel bajo de tensión, un "0" lógico, y se encuentra en reposo y por tanto desactivado, con un nivel alto de tensión, un "1" lógico. Además se deberá conectar los pines de VCC y GND a la fuente de alimentación correspondiente para el suministro de alimentación eléctrica, una fuente que sea capaz de suministrar la corriente necesaria para la excitación de las bobinas de los revés, siendo recomendable no usar la misma alimentación que Arduino, sobre todo si se alimenta a través de USB, ya que éste no dispone de una potencia suficiente y se producen comportamientos anómalos.

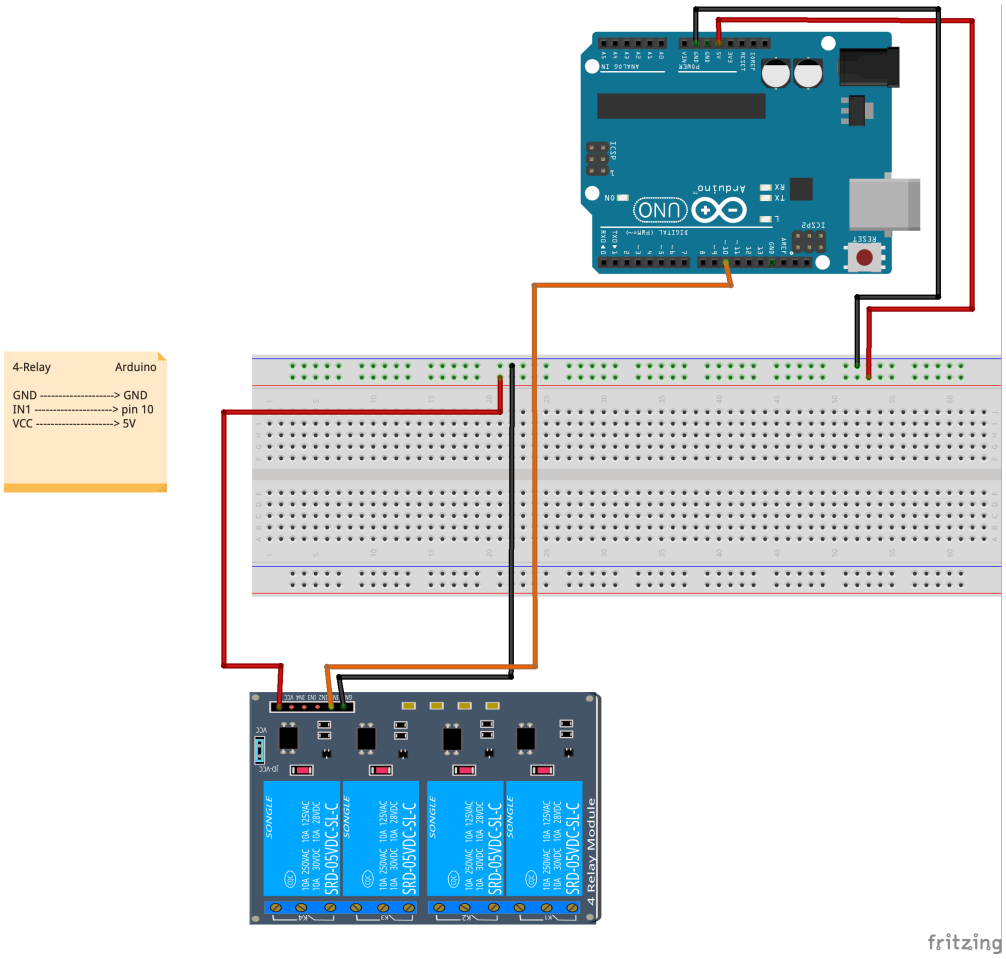


Imagen 47: Conexión Arduino y Módulo Relé de 4 canales

[Este componente no se pudo añadir finalmente al sistema, por falta de materiales y tiempo]

3.4.1.8 Minibomba de Agua Sumergible

A continuación se muestran las conexiones realizadas entre el módulo relé y las minibombas de agua. Para esta conexión es necesario incorporar una fuente de energía, en este caso al ser dos minibombas, se utilizan dos pilas cada una de 3.7 Volt.
Por lo tanto para nuestro prototipo se realizan las siguientes conexiones:
El lado positivo de la batería con los NO del módulo relé, el lado negativo de la batería con GND de la minibomba de agua y los COM del módulo relé con el VCC de la minibomba de agua.

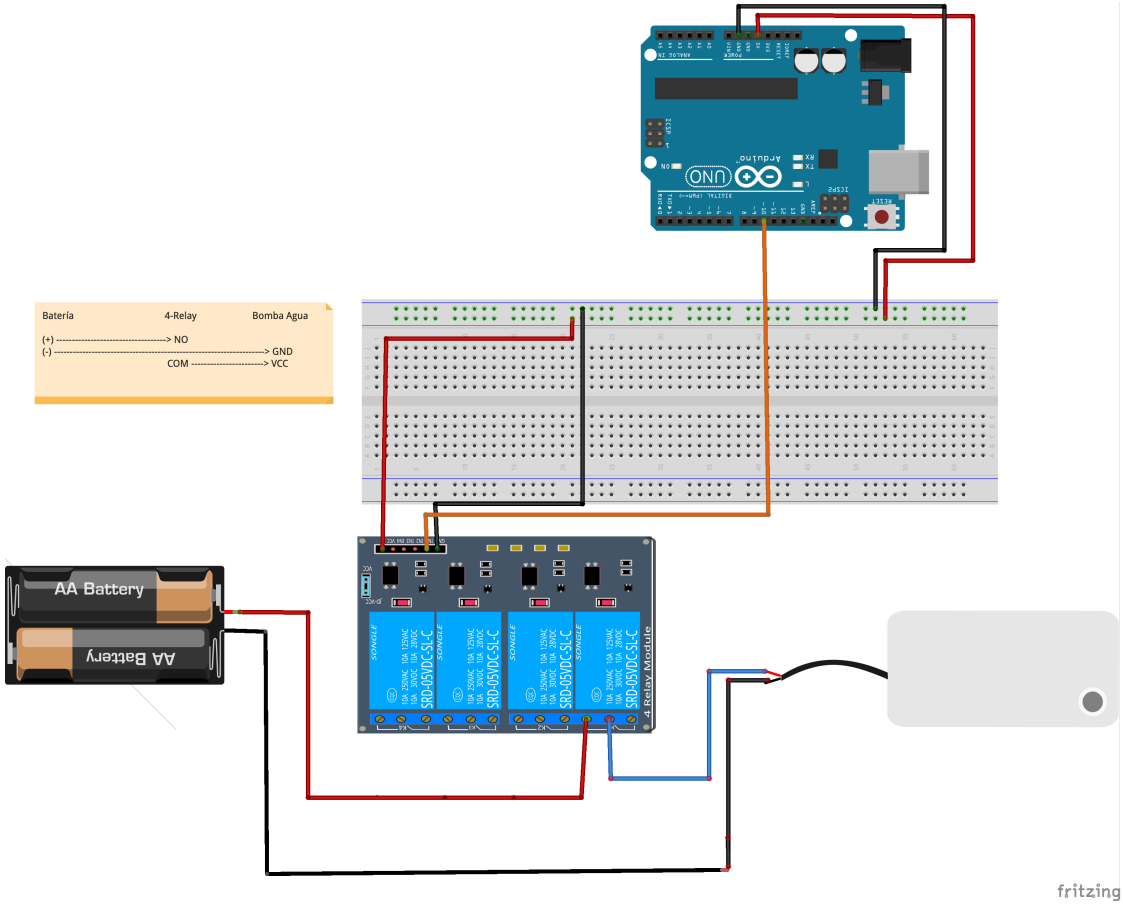
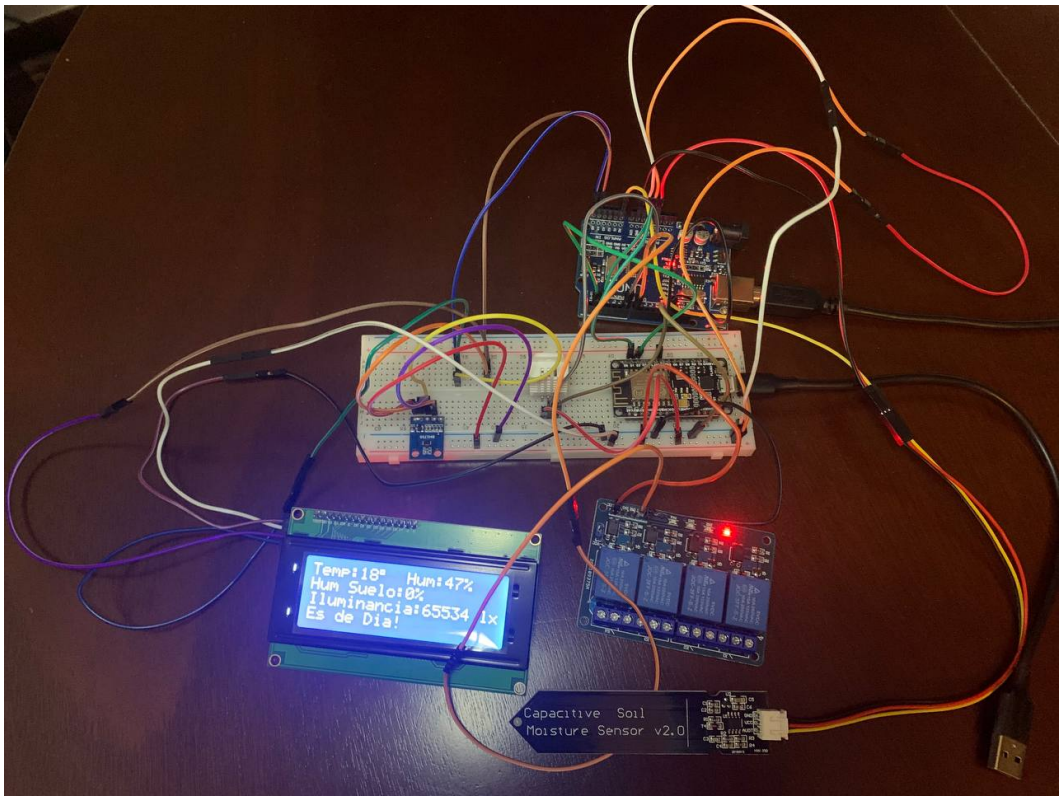


Imagen 48: Conexión Arduino, Relay, Baterías y Minibomba de Agua Sumergible

3.4.2 Prototipo

El sistema se ha implementado en un prototipo mediante una placa protoboard, donde se han realizado todas las conexiones de los componentes tal como se mostraron en el apartado anterior.
Los distintos componentes se fueron conectando y probando de manera independiente y posteriormente se realizó la integración del sistema completo.
Para que nuestro prototipo funcione correctamente, se procedió a realizar la compilación y cargado de nuestro código a la placa Arduino y al módulo NodeMCU por separado, utilizando el IDE de Arduino y realizando la conexión al PC mediante el puerto USB.
A continuación, se presentan imágenes del prototipo real realizado.



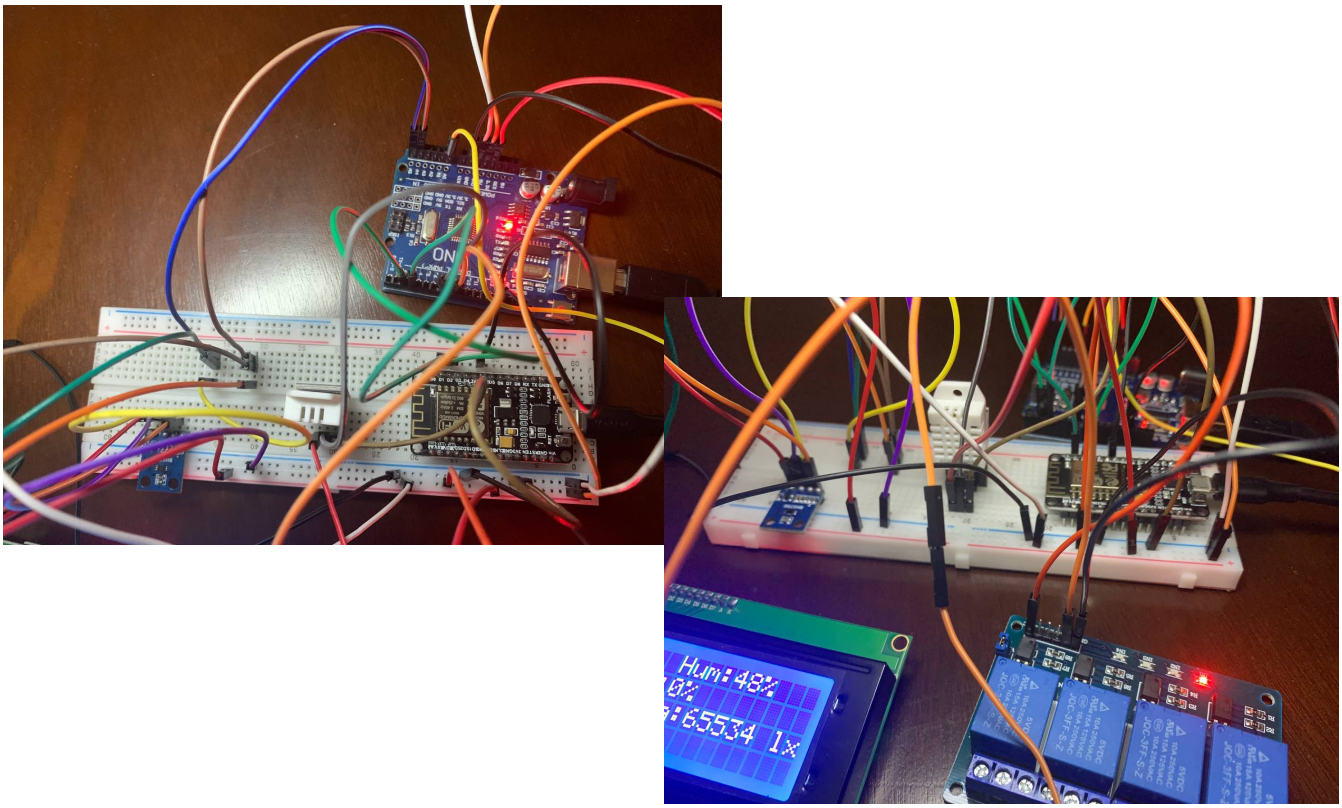


Imagen 49, 50 y 51: Imágenes del Prototipo

Algunas consideraciones del prototipo:
- Los sensores de temperatura, humedad y luminosidad, deberán estar situados en el exterior, pero protegido de las inclemencias del tiempo.
- El sensor de luminosidad, debe colocarse estratégicamente, de manera que permita la lectura de la intensidad de la luz.
- Los sensores de humedad del suelo, deberán colocarse en las macetas o en el huerto, insertados en la tierra.
- Los componentes que requieren comunicación I^2C, deberán estar situados relativamente cerca de la placa Arduino, ya que este protocolo de comunicación no permite disponer de un cableado excesivamente largo.

3.5 Presupuesto

A continuación, se detallan los costos de los diferentes componentes que se utilizaron para el desarrollo del diseño del prototipo del sistema de riego automatizado.
Los valores indicados son con IVA incluido.

Descripción de Recursos	Cantidad	Precio Unitario (UF)	Valor Total (UF)
Herramientas Tecnológicas			
Hardware	-	0 UF	0 UF
Software (Licencias libres)	-	0 UF	0 UF
Componentes			
Placa Arduino Uno R3	1	0,50 UF	0,50 UF
Protoboard	1	0,10 UF	0,10 UF
Sensor de Temperatura y Humedad DHT22	1	0,20 UF	0,20 UF
Sensor Capacitivo de Humedad de Suelo v1.2	1	0,13 UF	0,13 UF
Sensor de luminosidad BH1750	1	0,10 UF	0,10 UF
Módulo Relé de 4 canales	1	0,13 UF	0,13 UF
Node MCU Wifi ESP8266	1	0,29 UF	0,29 UF
Mini Bomba de Agua Sumergible	1	0,12 UF	0,12 UF
Pantalla Display LCD 20x4	1	0,23 UF	0,23 UF
Porta Batería	1	0,050 UF	0,050 UF
Pilas 3.7 V	2	0,10 UF	0,20 UF
Cableado	1	0,067 UF	0,067 UF
TOTAL			2,117 UF

Tabla N° 6: Presupuesto Sistema de Riego

El presupuesto final es de 2,117 UF que equivalen aproximadamente a $63.031 pesos chilenos. Lo que comparado con otros sistemas de riego o estaciones de medición de temperatura y humedad de similares características supone un costo bastante bajo.
Hay que considerar que el valor del prototipo considera la medición para una maceta o planta. En el caso que se requiera el uso del prototipo para más plantas, se deberán agregar más Sensores Capacitivos de Humedad de Suelo y Mini Bombas de Agua.

Para abaratar costos, se podrían adquirir los componentes al por mayor, lo que bajaría considerablemente el valor final por producto.

3.6 Pruebas finales y Resultados

Se realizaron distintas pruebas para revisar el comportamiento del prototipo, verificando que haya un correcto funcionamiento tanto del hardware como del software.
Antes de comenzar con la realización de las pruebas, se procedió a realizar la calibración de los sensores de humedad del suelo para no afectar los resultados del prototipo.
En cuanto a las pruebas, se ha verificado que los datos recibidos por los sensores son coherentes con el tipo de medida y magnitud del parámetro a medir para cada uno de ellos. Se ha comprobado que la transferencia de datos entre Arduino y el módulo WiFi es consistente y sincronizada, permitiendo el envío y recepción de los mensajes de comunicación.
Además, se han realizado distintas pruebas para verificar el funcionamiento de las interfaces de usuario, como la pantalla LCD y la aplicación móvil, comprobando que la interacción con el usuario es satisfactoria y que el envío de ordenes hacia los actuadores funcione de manera correcta.

3.6.1 Calibración Sensor Capacitivo de Humedad de Suelo

Para el correcto uso del sensor de humedad de suelo, es necesario realizar primero una calibración del sensor.
Para efectuar esta calibración, se realizó un pequeño programa en Arduino (ver en Anexo) con el cual se puede obtener los valores obtenidos del sensor en distintas condiciones de humedad. Además del programa en Arduino, se debe considerar los siguientes pasos:

1º Abrir el monitor serial y colocar los baudios a 9600.
2º Revisar el valor que tiene el sensor cuando aun no es colocado en la superficie a medir la humedad, para que registremos una Humedad = 0% (Valor1).
3º Insertar el sensor dentro de un vaso con agua, sin pasar de la linea roja o de la profundidad máxima.
4º Revisar en el monitor serial el valor que se registró para tener el valor de la Humedad = 100% (Valor2).

En nuestro prototipo, se obtuvo un Valor1=579 y un Valor2=220, por lo cual tendremos un rango de valor de:

- 579-526 = 0% humedad, suelo seco
- 526-250 =50% humedad, suelo húmedo
- 250-220=100% humedad, suelo encharcado

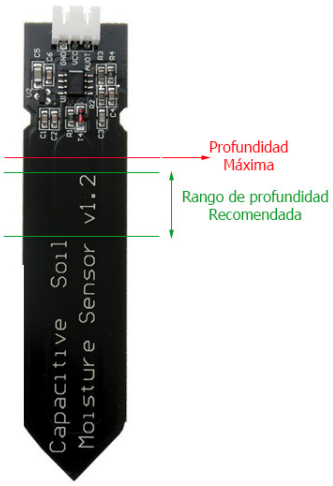


Imagen 52: Rango de Calibración, Sensor Capacitivo de Humedad de Suelo

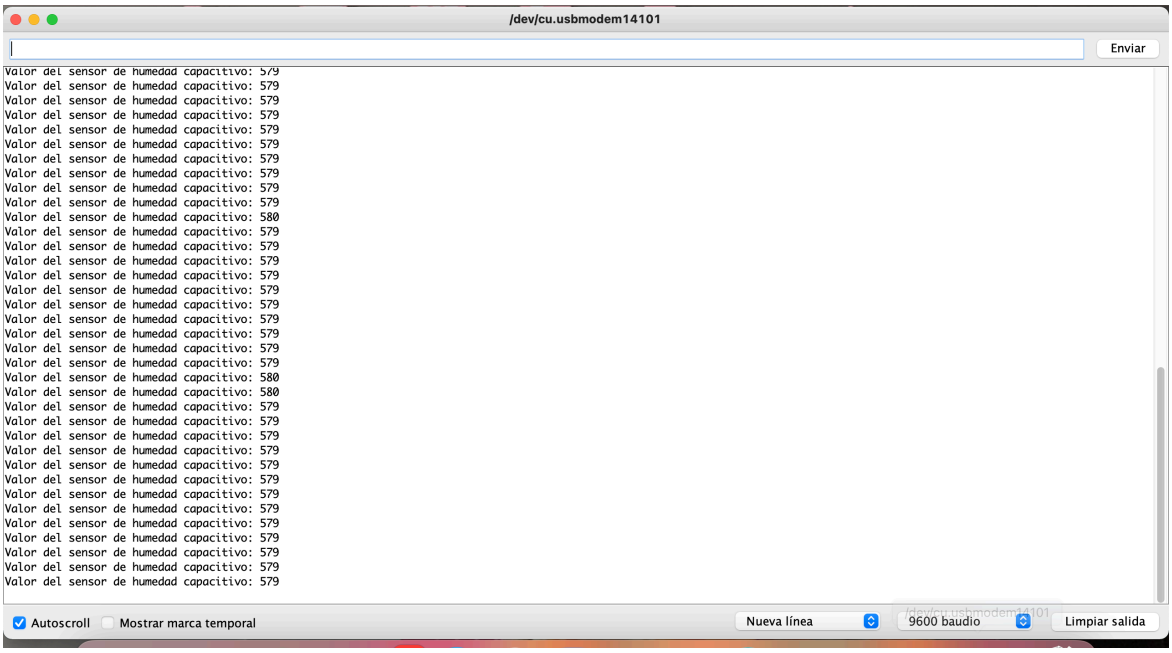


Imagen 53: Calibración Sensor Capacitivo de Humedad de Suelo en suelo seco

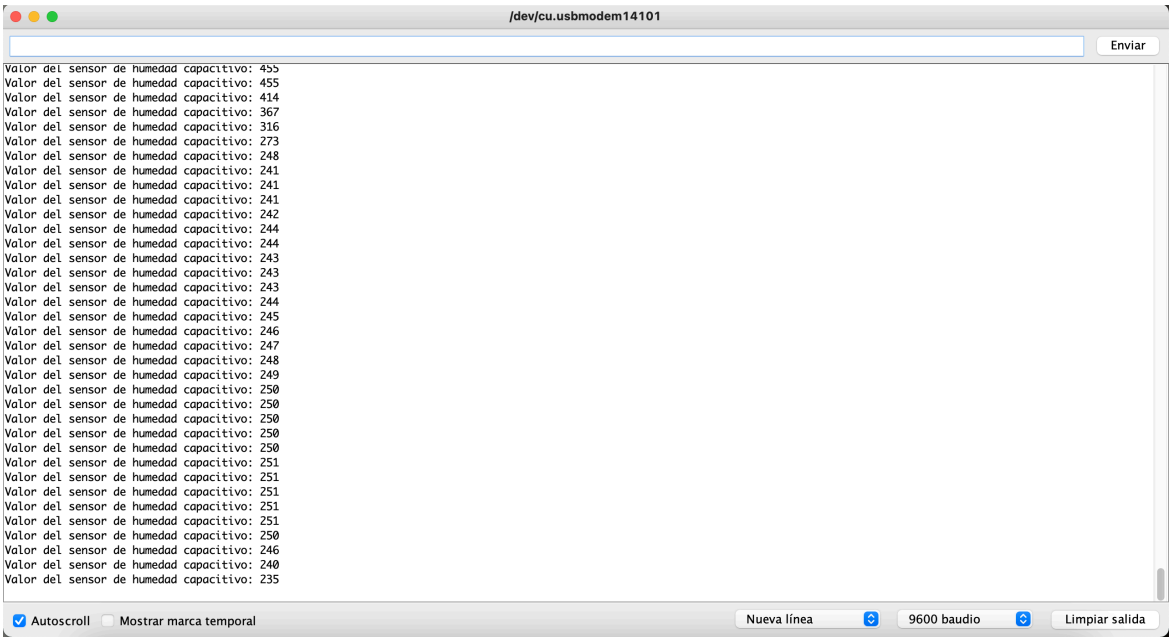


Imagen 54: Calibración Sensor Capacitivo de Humedad de Suelo en suelo húmedo

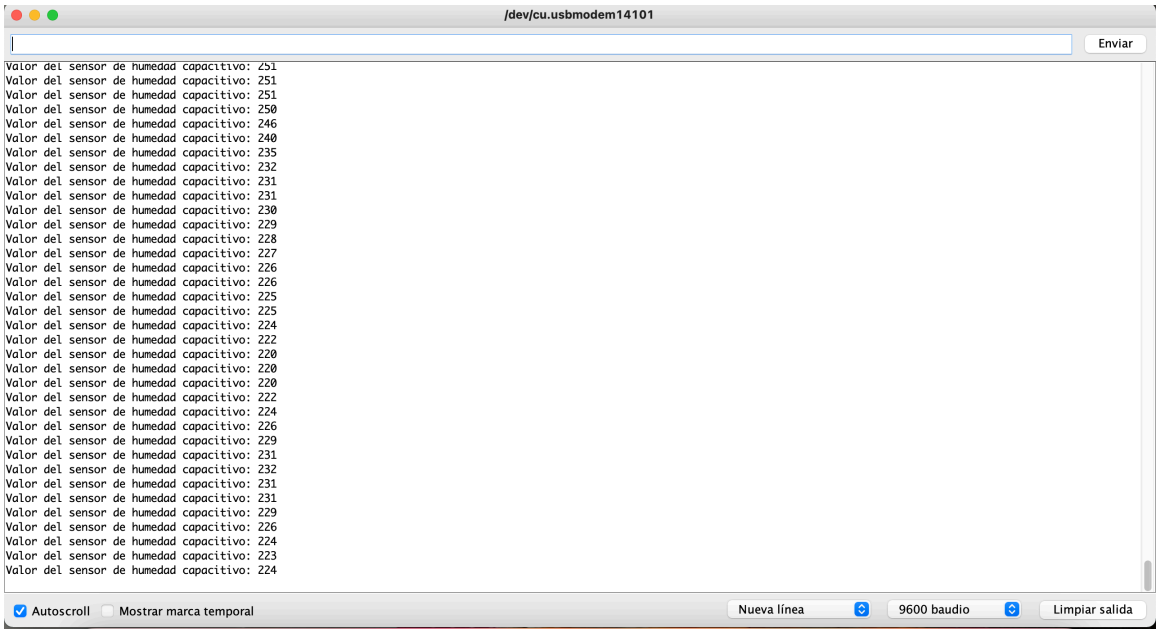


Imagen 55: Calibración Sensor Capacitivo de Humedad de Suelo en suelo encharcado

3.6.2 Verificación de datos de los sensores

Se han realizado distintas verificaciones de la recepción de datos desde los sensores, para determinar si estos datos son consistentes a los valores esperados para cada una de las magnitudes, ya sea humedad de suelo, de aire, temperatura, iluminancia. Teniendo en cuenta el entorno y las condiciones climáticas de donde están ubicados los sensores.

Estas verificaciones se pueden visualizar gracias al IDE de Arduino, pudiendo obtener los valores de los sensores en tiempo real, como se aprecia en la siguiente imagen.

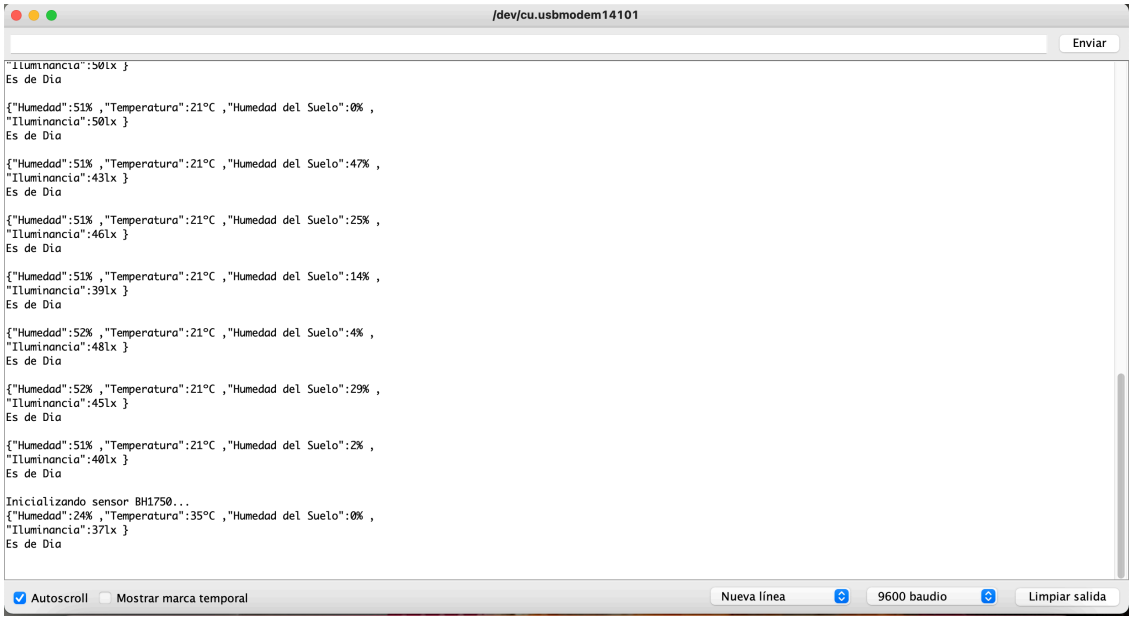


Imagen 56: Prueba recepción de datos desde sensores

3.6.3 Pruebas regulación luminosidad

Se realizan las mediciones de la intensidad de iluminancia mostrando los resultados medidos en la consola de la placa Arduino y también en la pantalla LCD.
Dependiendo del nivel de iluminancia y los ejemplos de la tabla nº5 podremos determinar si es de noche o si hay demasiada intensidad de luz para finalmente decidir si es beneficioso o no realizar el regado con esas intensidades de luz.
Para las pruebas se procedió a iluminar con una linterna para dar luz de dia y a tapar el sensor para hacer el efecto de noche.

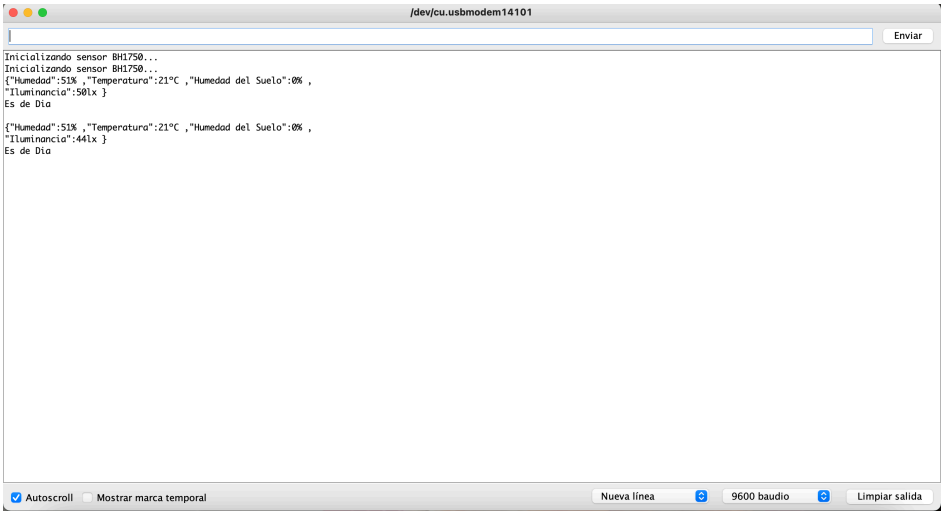


Imagen 57: Resultados inicialización sensor BH1750





Imagen 58 y 59: Resultados medición sensor BH1750 en pantalla LCD

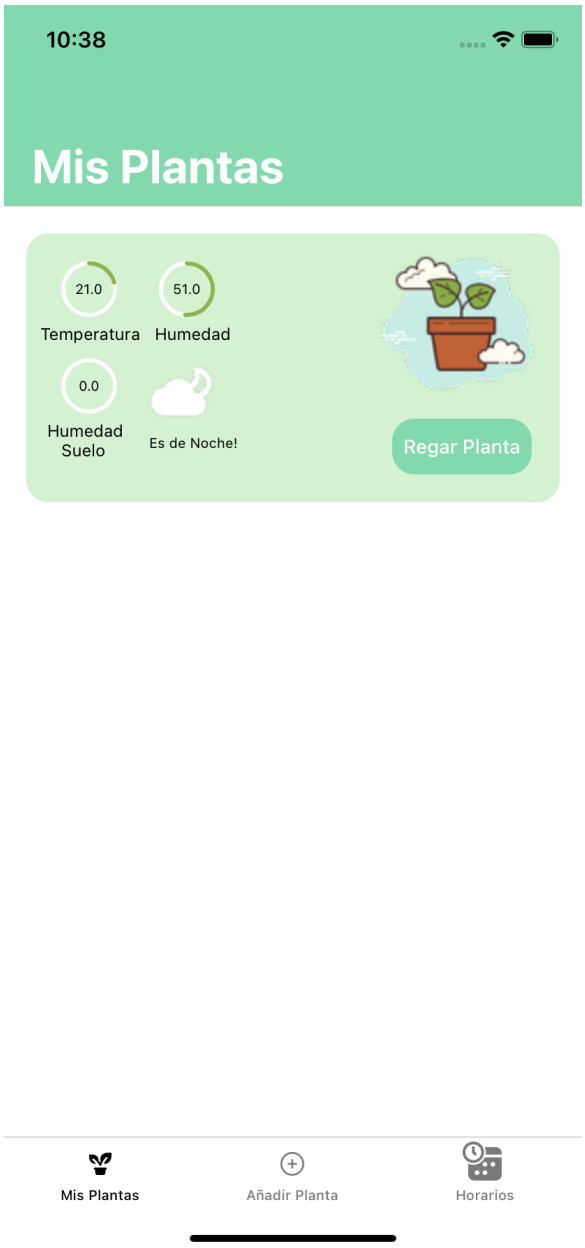


Imagen 60: Resultados medición sensor BH1750 en aplicación móvil

3.6.4 Monitorización de condiciones ambientales

Para verificar que los datos entregados por los distintos sensores son coherentes con el entorno en el que se encuentran y que además se estén actualizando en tiempo real, se procederá a tomar los estados iniciales de los sensores, para posteriormente modificar las condiciones de cada sensor.

Estado de condiciones medioambientales iniciales:

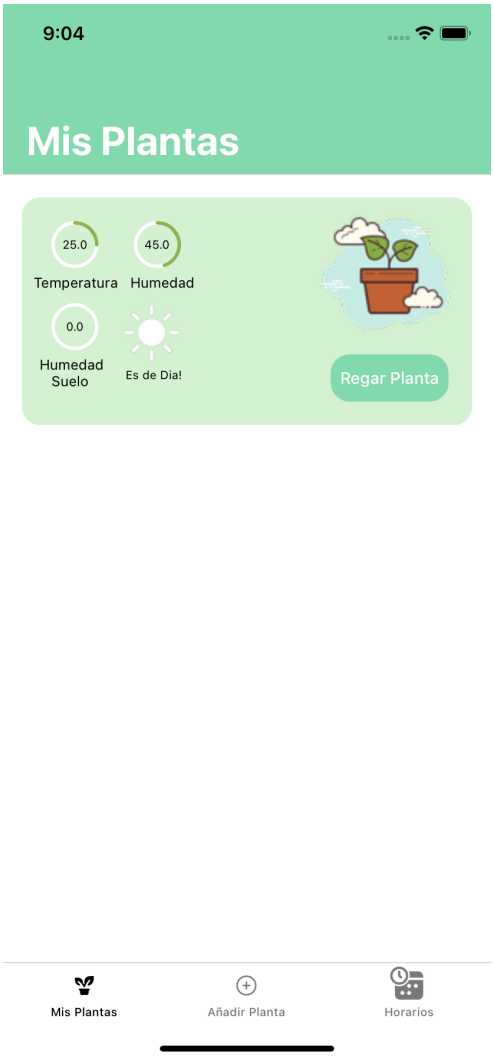


Imagen 61: Aplicación Móvil: Estado de condiciones medioambientales iniciales



Imagen 62: LCD: Estado de condiciones medioambientales iniciales

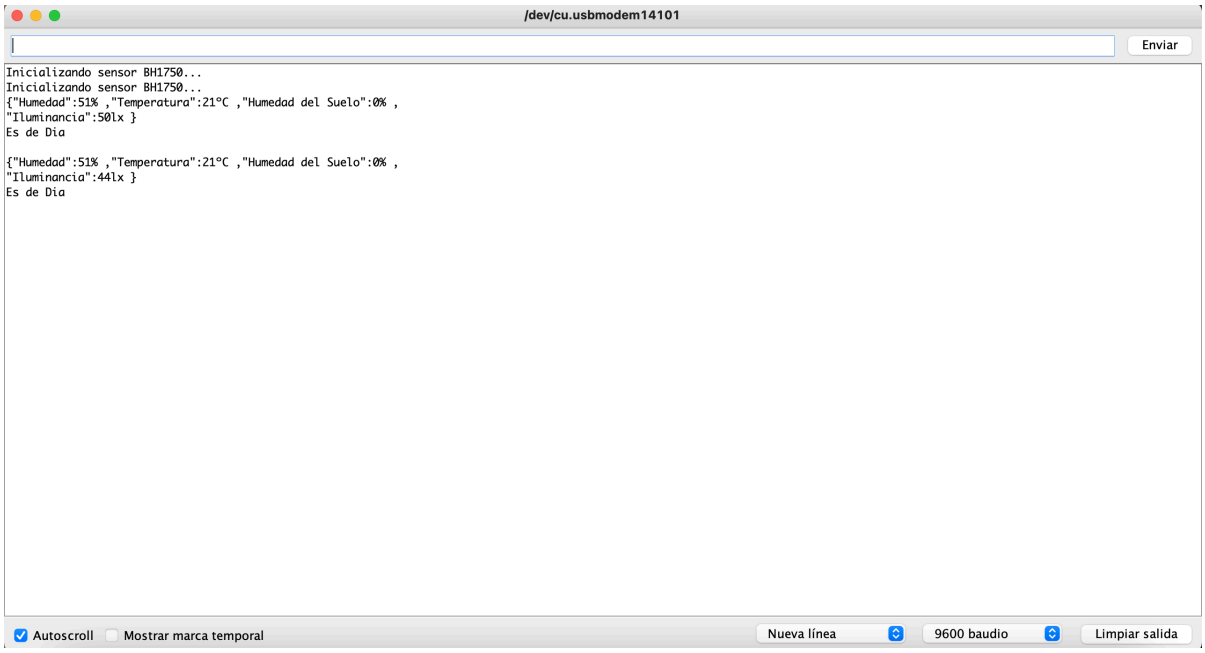


Imagen 63: IDE Arduino: Estado de condiciones medioambientales iniciales

Sensor de humedad de suelo:
Para poder establecer un cambio del estado inicial, se procedió a introducir el sensor de humedad de suelo en un entorno de humedad distinto al que se encontraba.

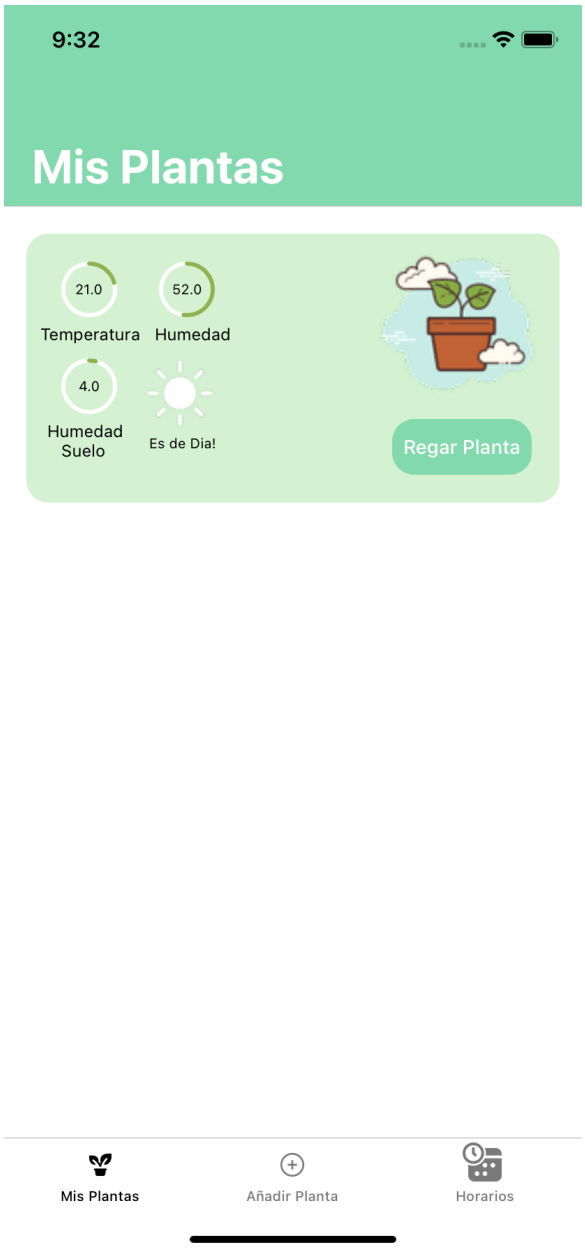


Imagen 64: Aplicación Móvil: Prueba Cambio Estado humedad de suelo.

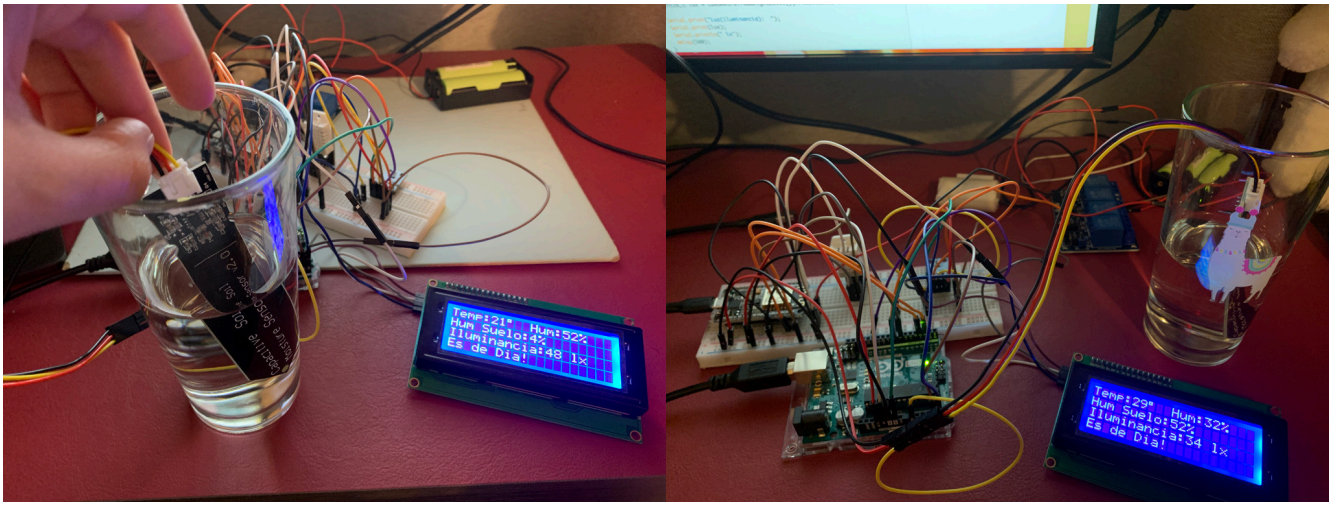


Imagen 65 y 66: LCD: Prueba Cambio Estado humedad de suelo.



Imagen 67: IDE Arduino: Prueba Cambio Estado humedad de suelo.

[A partir de las siguientes pruebas, el sensor de iluminancia comenzó a fallar, por lo que los valores de iluminancia marcarán siempre un valor de 65534 lx.]

Sensor de temperatura y humedad ambiental:
Para poder establecer un cambio del estado inicial, se procedió a calentar el sensor con un secador de cabello, el cual permite introducir aire caliente disminuyendo la humedad y aumentando la temperatura. Estos cambios se pueden visualizar en la pantalla LCD y Aplicación Móvil.

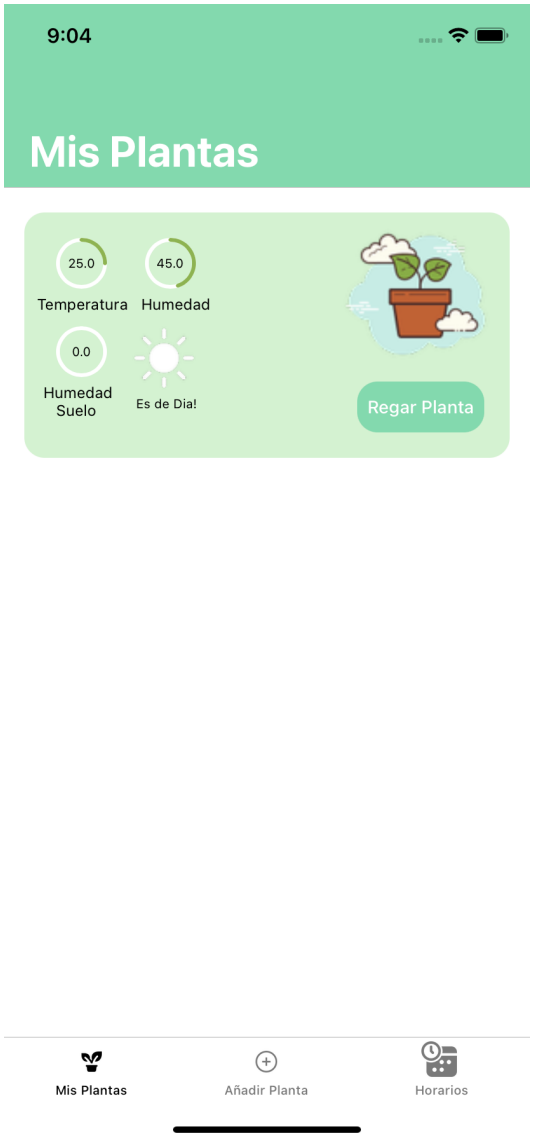


Imagen 68: Aplicación Móvil: Prueba Cambio Estado temperatura y humedad ambiental.

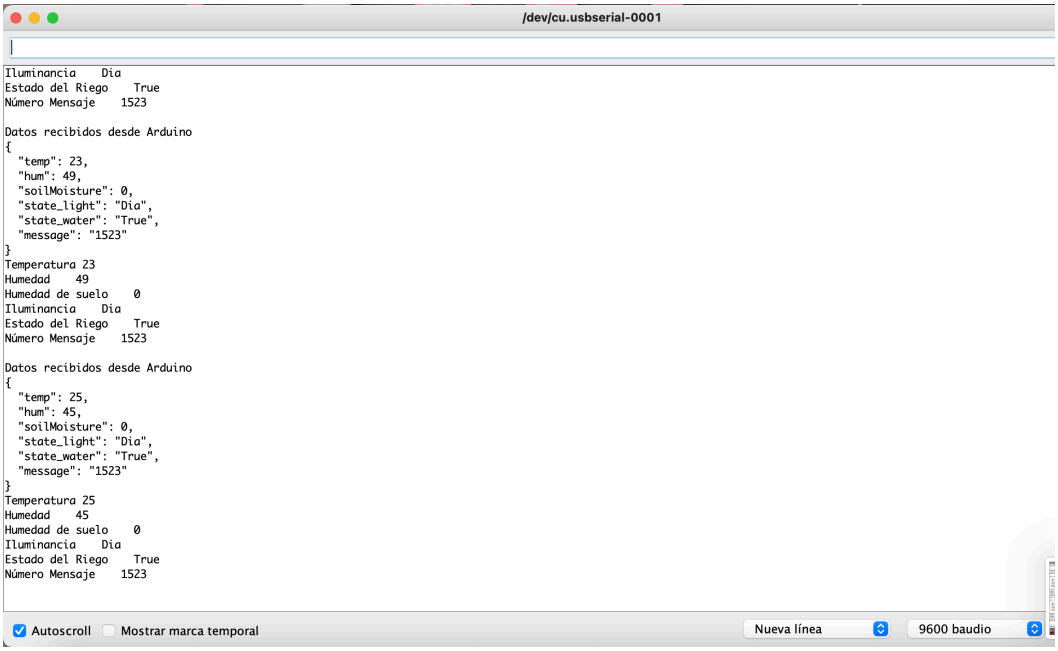


Imagen 69: IDE Arduino: Prueba Cambio Estado temperatura y humedad ambiental.

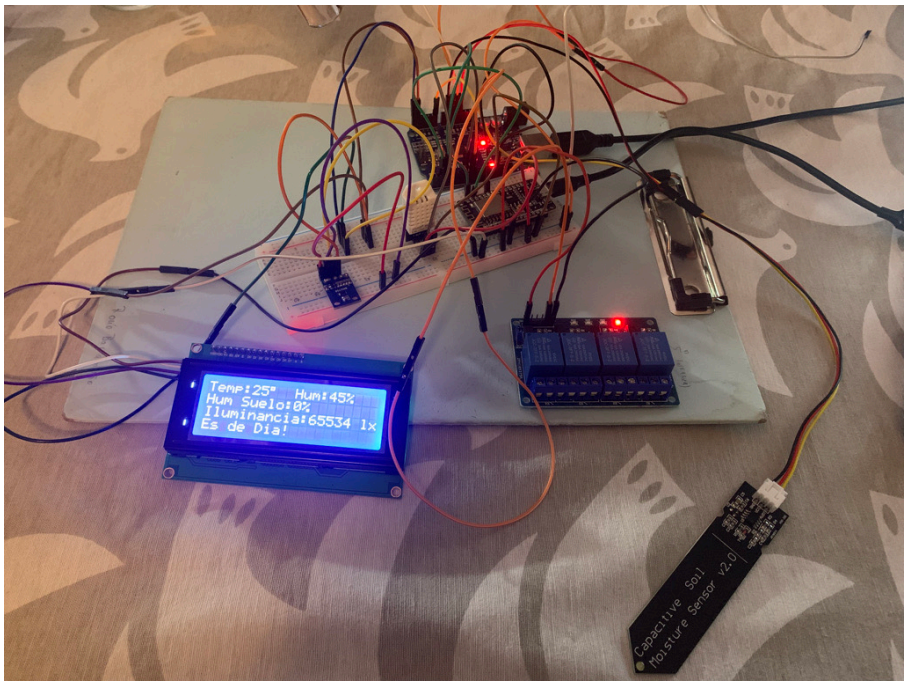


Imagen 70: LCD: Prueba Cambio Estado temperatura y humedad ambiental.

3.6.5 Verificación de comunicación entre Arduino y Módulo WiFi

Para verificar el correcto funcionamiento y la sincronización de la comunicación entre ambos componentes, se han programado mensajes que nos permitan visualizar de manera intuitiva y mediante la pantalla del IDE de Arduino.
Lo primero que se debe verificar es que nuestro módulo WiFi se haya conectado de manera exitosa a nuestra red WiFi.

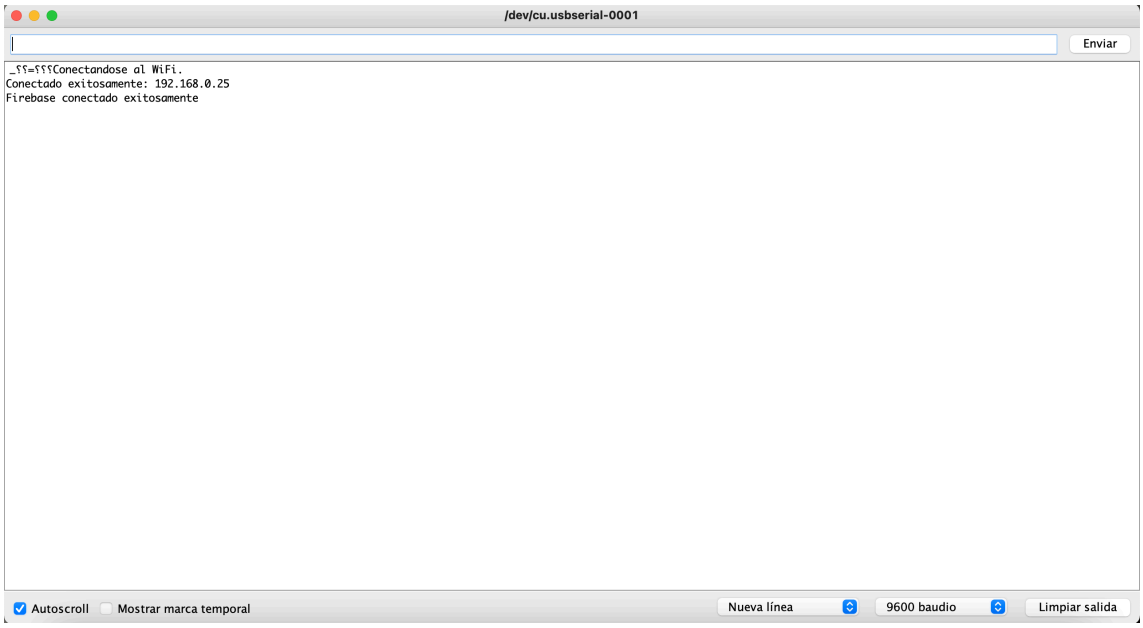


Imagen 71: Prueba de conexión módulo WiFi con red WiFi

Lo siguiente que se debe verificar es que los datos que recopilamos de nuestros sensores hayan podido llegar hasta nuestro módulo WiFi de forma exitosa, coherente y de manera sincronizada. Para esto identificaremos con un número el mensaje, para nuestro ejemplo a continuación será el número 1523. Con esto verificaremos que la comunicación entre Arduino y nuestro módulo WiFi se está realizando de manera correcta y sincronizada.

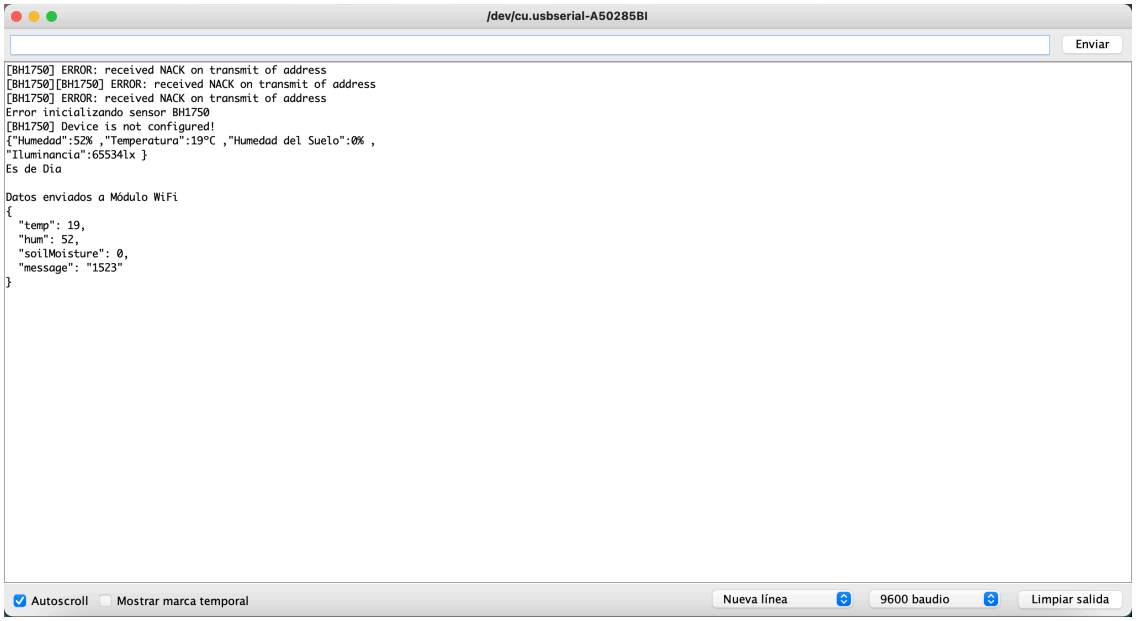


Imagen 72: Prueba de envío de datos desde Arduino al módulo WiFi

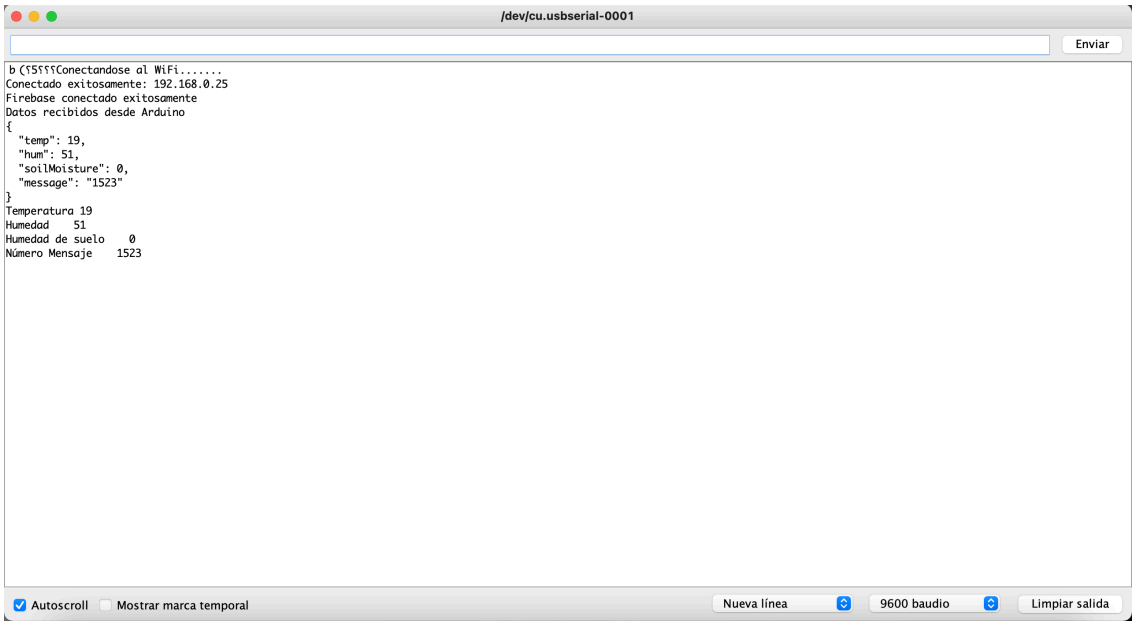


Imagen 73: Prueba de recepción de datos en módulo WiFi

3.6.6 Verificación de recepción de datos en la base de datos de Firebase

Como vimos en el apartado 3.6.2, se pudo verificar que los datos provenientes de los sensores eran correctos, y además en el apartado anterior, se pudo verificar que existe una correcta comunicación entre Arduino y el módulo WiFi, por lo que ahora nos corresponde verificar que los datos que recibimos correctamente en nuestro módulo WiFi, puedan ser enviados a la base de datos de tiempo real de Firebase.
Para esto primero tenemos que verificar que la conexión a Firebase se esté realizando de manera correcta y luego verificar que los datos se están enviando a la base de datos de Firebase de forma exitosa.
Mediante el IDE de Arduino, podemos visualizar el mensaje de que se estableció de manera correcta la comunicación con Firebase. Y mediante la plataforma de Firebase podemos revisar los datos que llegaron a la base de datos.

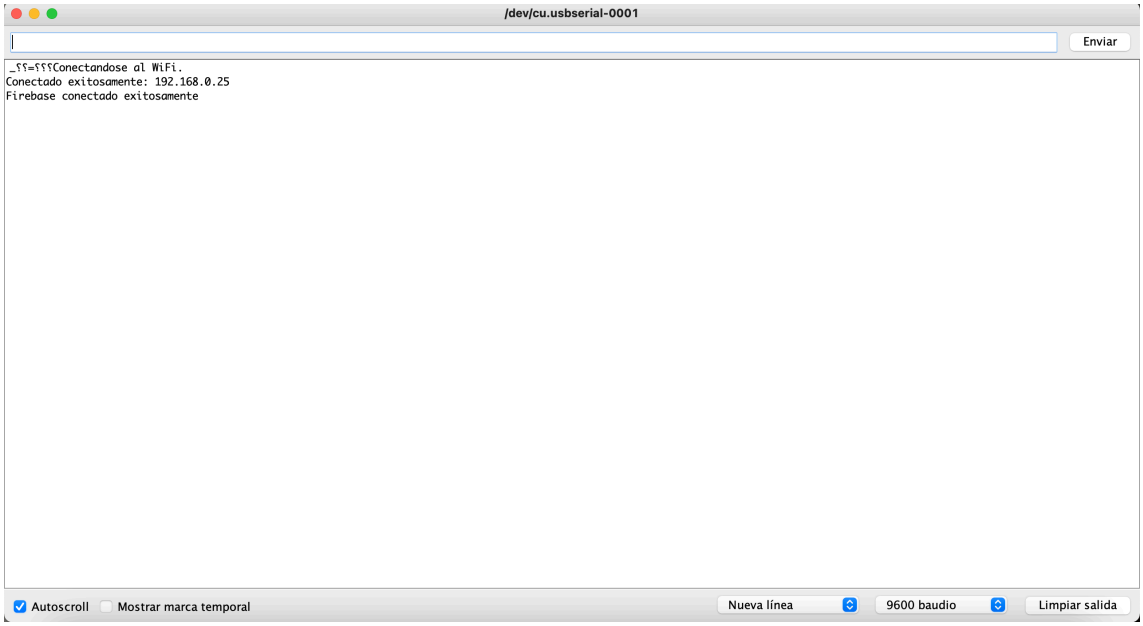
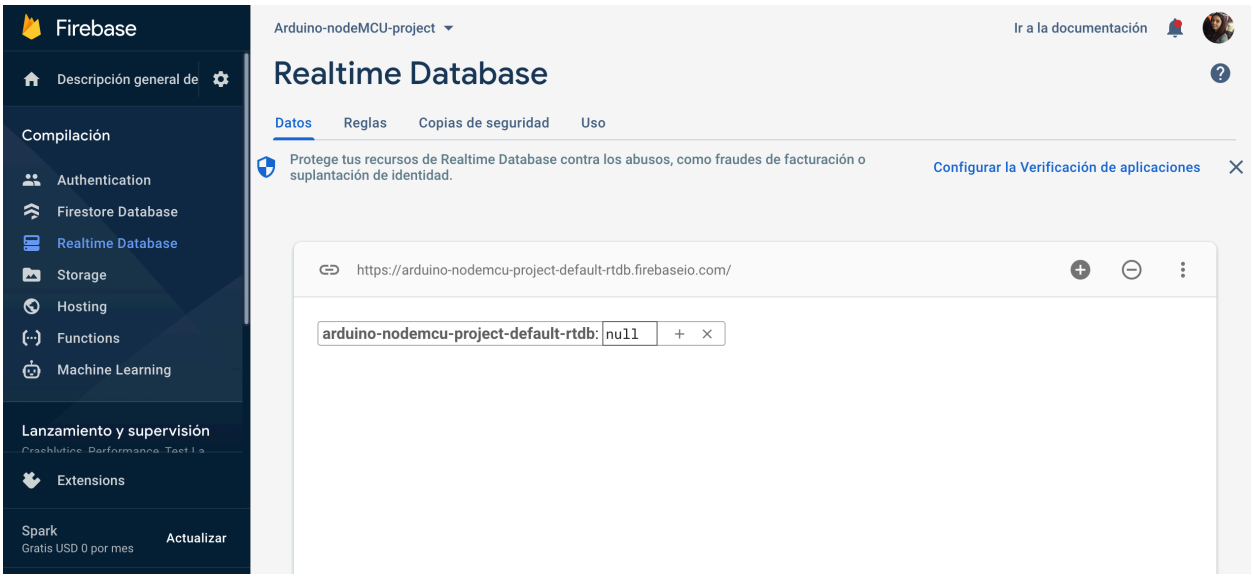


Imagen 74: Prueba de conexión con Firebase



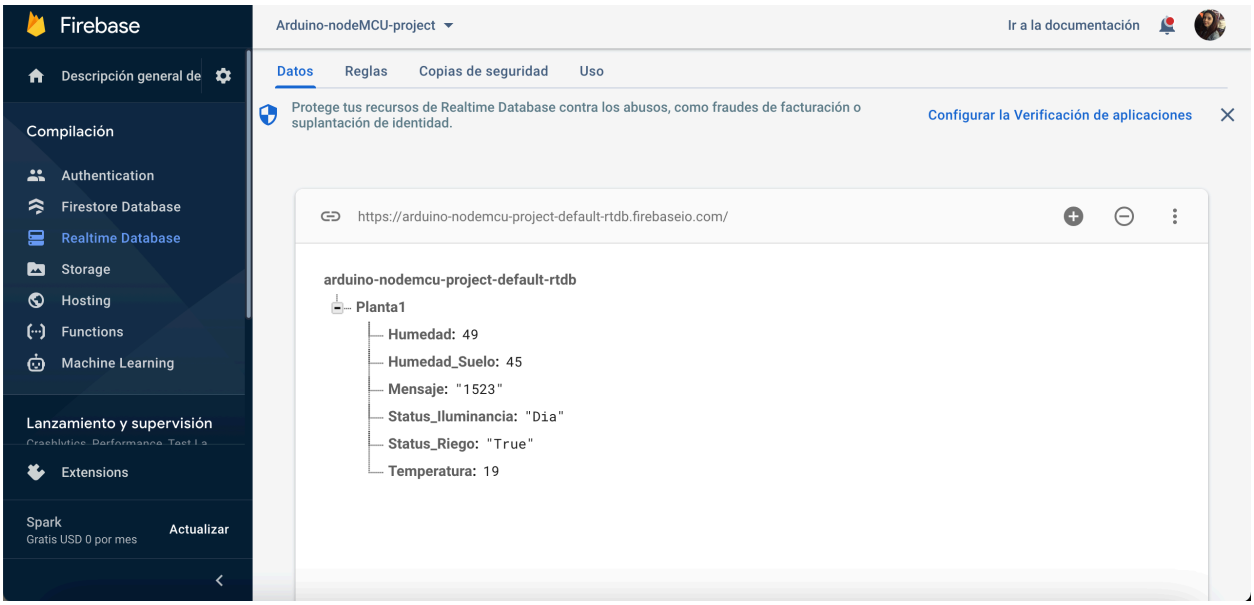


Imagen 75 y 76: Prueba recepción de datos en base de datos de Firebase

3.6.7 Prueba Activación de riego desde Aplicación Móvil

Para verificar el funcionamiento de la activación de riego desde la aplicación móvil, se realizaron los siguientes pasos:
Activación de riego:
- Se accede a la aplicación móvil y seleccionando el botón verde “Regar Planta” se procede a enviar los datos hacia los actuadores y activar las bombas de agua.
Posteriormente se comprueba que el indicador LED del relé está encendido y por lo tanto se activa el riego.

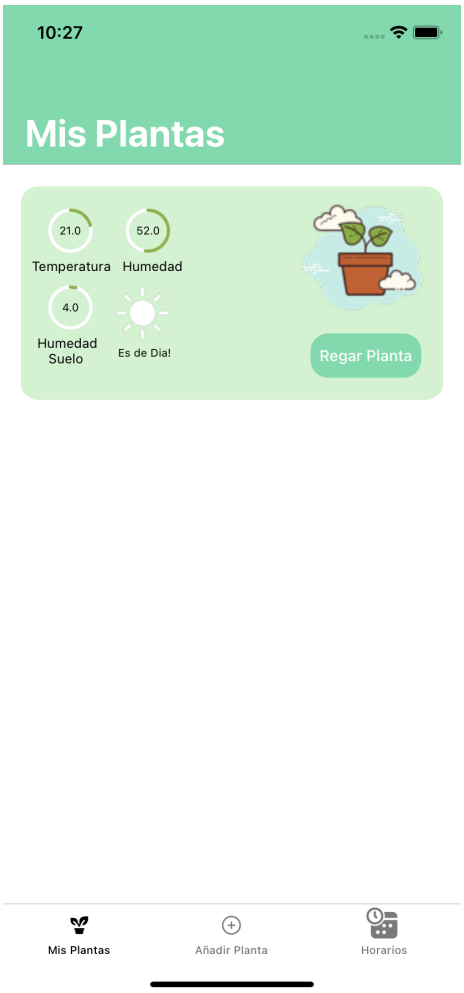


Imagen 77: Prueba Activación de riego desde Aplicación Móvil

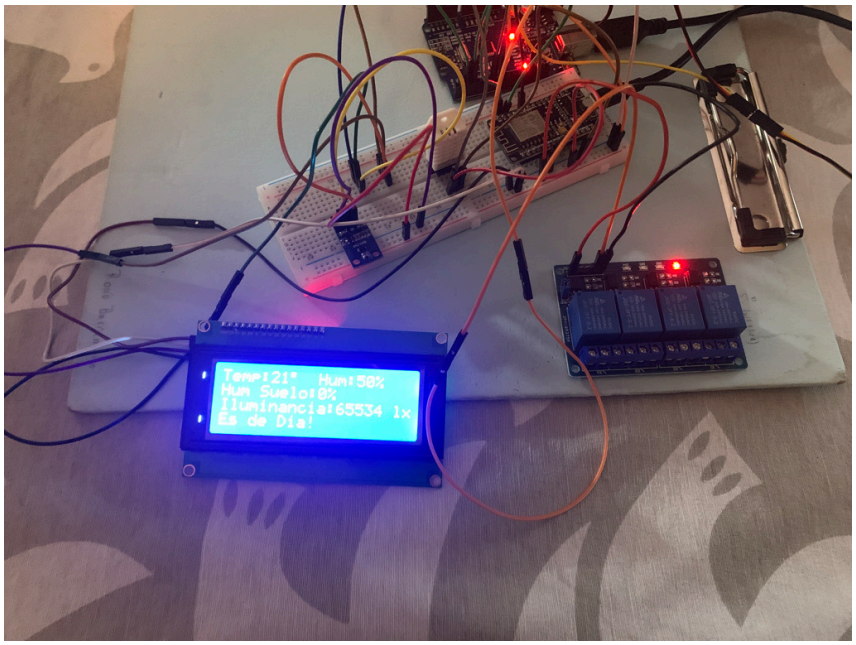


Imagen 78: Relé activado

Desactivación de riego:
- Se accede a la aplicación móvil y seleccionando el botón rojo “Desactivar Riego” se procede a enviar los datos hacia los actuadores y desactivar las bombas de agua. Posteriormente se comprueba que el indicador LED del relé está apagado y por lo tanto se desactiva el riego.

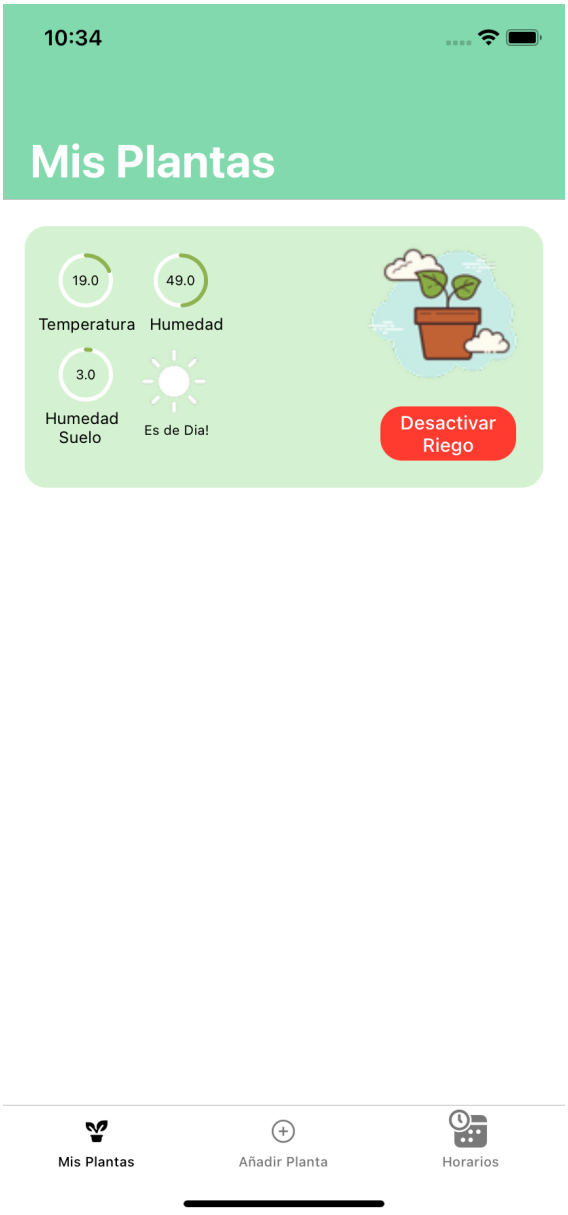


Imagen 79: Prueba Desactivación de riego desde Aplicación Móvil

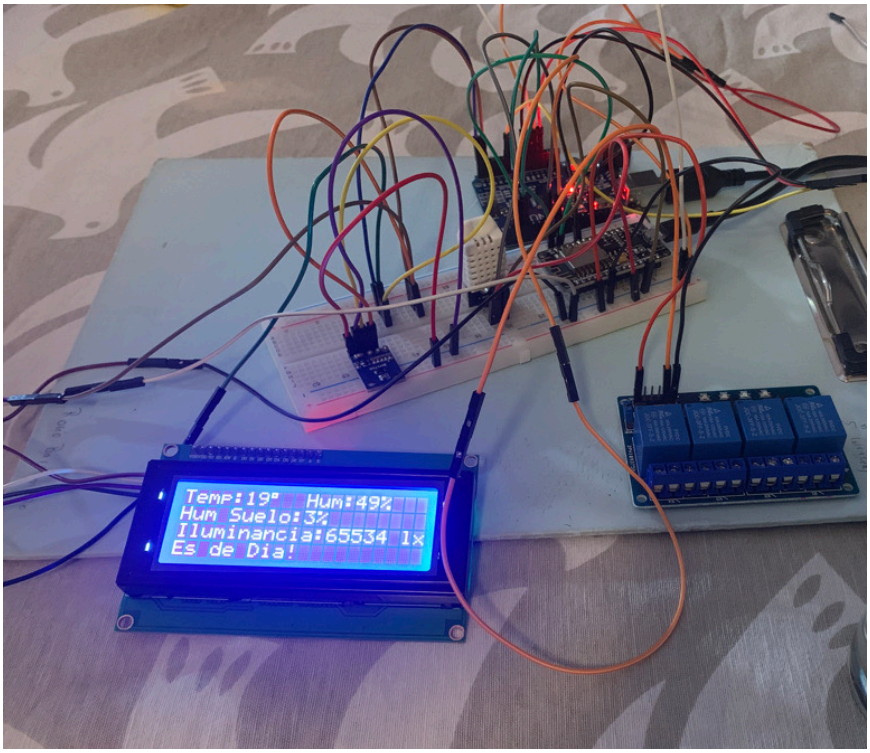


Imagen 80: Prueba Relé desactivado

CAPÍTULO IV
CONCLUSIONES

RESUMEN Y CONCLUSIONES
TRABAJO FUTURO
POSIBILIDAD PRODUCTO COMERCIAL

4.1 Resumen y Conclusiones

En este estudio se ha diseñado, implementado y probado un sistema de riego automático de bajo costo para un huerto urbano. Este sistema está compuesto por una serie de sensores que van registrando los datos ambientales y un microcontrolador que analiza los datos y determina la necesidad del riego activando o desactivando los relés. Esta activación o desactivación se puede realizar en la aplicación móvil de manera muy intuitiva.
Para la comunicación inalámbrica de este sistema, se utilizó un módulo WiFi que permite enviar y recibir los datos que obtenemos de nuestro microcontrolador hacia y desde la base de datos en tiempo real de Firebase, la cual se consulta desde la aplicación móvil para finalmente mostrar los datos.

Para el diseño del sistema, fue necesario determinar qué protocolos de comunicación se utilizan en los distintos componentes, para realizar correctamente las conexiones en los pines del microcontrolador. Estos diseños se pudieron replicar gracias al programa Fritzing, el cual es opensource y muy fácil de utilizar. Sin embargo algunos de los componentes utilizados en el sistema no venían incluidos en el programa Fritzing, por lo que se tuvo que descargar estos componentes para poder importarlos en el programa.

En el prototipo se pudieron obtener resultados positivos respecto a las mediciones de humedad ambiental, temperatura y humedad de suelo. Sin embargo las mediciones de iluminancia fallaron en las últimas pruebas, se desconoce la razón de la falla, pero pudo haber sido por una incorrecta soldadura en el sensor o puede que no sea un sensor 100% confiable. Por lo que a futuro, se debería reemplazar este sensor de iluminancia por otro que nos entregue mayor seguridad y tolerancia a fallos.
Además se pudo comprobar que la activación y desactivación del relay en el prototipo funcionó de manera esperada, ya que al obtener un bajo porcentaje de humedad de suelo se activaba el relay y en caso contrario se desactivaba. Esta activación y desactivación, permitiría o no el paso del agua a través de las minibombas de agua realizando el riego en las plantas. Lamentablemente no se pudo concretar la integración de las minibombas de agua al prototipo, por falta de materiales y tiempo, pero a futuro esto sería una mejora al sistema.
También se pudo concretar en el prototipo una correcta comunicación entre el microcontrolador y el módulo WiFi, pudiendo enviar y recibir constantemente los datos entre ambos componentes.
Se ha conseguido también la creación de una aplicación móvil en iOS, que muestra el

estado del cultivo mediante la visualización de las mediciones en la pantalla de un equipo móvil o tablet y que además permite activar o desactivar el riego desde la misma aplicación.
En general, el prototipo cumple todos los objetivos propuestos al inicio de este estudio. Las pruebas realizadas demuestran que el prototipo funciona adecuadamente según las especificaciones propuestas y la implementación del sistema permitiría realizar de manera satisfactoria el riego automático en un huerto urbano, con un uso eficiente del agua basado en el balance de la humedad del suelo y temperatura ambiental, de bajo costo y sencillo de utilizar.

El sistema de riego automático permitirá mejorar el control del riego, reduciendo el consumo de agua y mejorando la productividad en los cultivos. Sin embargo, actualmente, el sistema está limitado a espacios reducidos, aunque es posible ampliarlo, manteniendo su diseño, pero integrando accesorios de mayor potencia en áreas de cultivos más grandes.

4.2 Trabajo Futuro

El diseño propuesto puede ser escalable y mejorado con las siguientes propuestas:

- **Incorporar Bombas de Agua:** Por factores de tiempo no se pudo completar el agregar las mini bombas de agua sumergibles al sistema, pero siguiendo el esquema del diagrama de conexión se podrían añadir para poder bombear agua hacia las plantas.

- **Cambio de sensor de iluminancia:** Debido a que el sensor de iluminancia terminó fallando, ya sea porque se realizó una incorrecta soldadura o porque a lo mejor este sensor no es 100% confiable, se debería probar otros tipos de sensores de iluminancia para añadirlos al sistema.

- **Incorporar Fuente de Energía:** Para que el sistema funcione de manera autónoma, se puede incorporar una fuente de energía, con la cual se suministrará energía a la placa de Arduino para su funcionamiento, como por ejemplo una batería externa de 5V.

- **Caja con protección para la intemperie:** Para que el sistema sea más compacto y esté más ordenado en cuanto al cableado, se podrían acomodar todos los sensores y componentes del sistema en una caja, la cual también brindaría protección al sistema de las inclemencias del exterior. Esta caja se podría confeccionar o se podría reutilizar los modelos que ya existen para imprimirla en 3D.

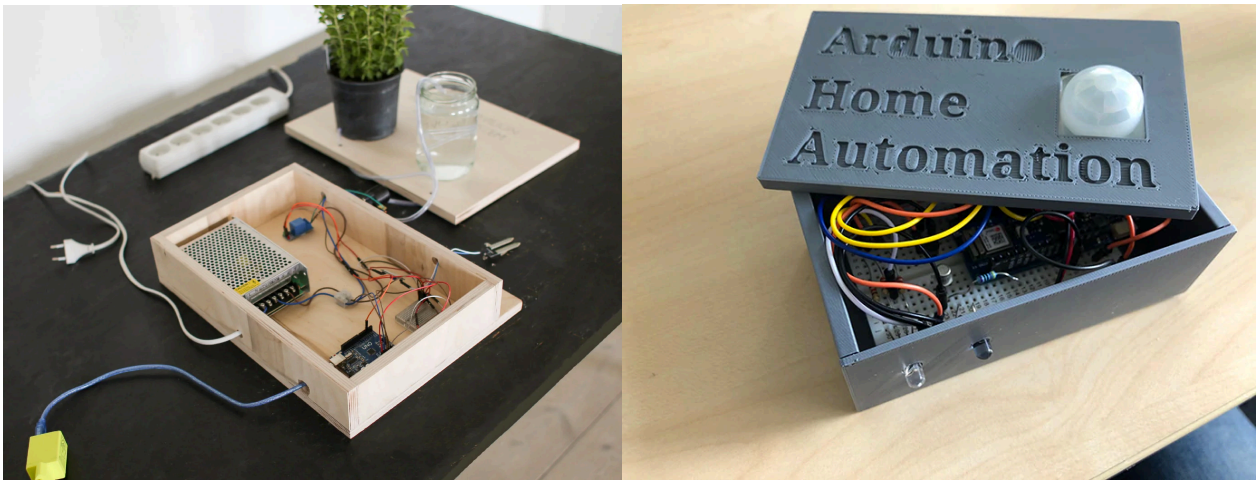


Imagen 81 y 82: Ejemplos de caja para protección del sistema

- **Almacenamiento de datos de los sensores en una tarjeta micro SD:** Para poder almacenar los datos en un tarjeta micro SD, se necesitará incorporar en el prototipo un Lector de tarjeta micro SD, junto con una tarjeta micro SD con suficiente espacio para guardar los datos. Esto sería beneficioso como un sistema de respaldo de datos y en caso de cortes de energía o fallas en la fuente de energía.

- **Continuar el Desarrollo de la Aplicación Móvil:** Se podría continuar trabajando en las secciones “Añadir Planta” y “Horarios” de la aplicación móvil.
En la sección “Añadir Planta”, se podría personalizar el nombre de la planta, añadir categoría de planta, la especie, etc. para poder monitorizar a más plantas en el sistema.
En la sección “Horarios", se podría determinar horarios de riego, fecha de aplicación de fertilizantes, etc.
Además se podría incluir notificaciones, para que se pueda advertir al usuario en caso de falta de riego o cuando corresponda el riego de acuerdo a los horarios definidos, etc.

- **Integrar domótica:** Actualmente hay cada vez más hogares que disponen de sistemas domóticos para la gestión integral de la vivienda. Home Assistant es una plataforma de domótica de código abierto diseñada para integrar y controlar todos los dispositivos del hogar y automatizar su control. Esta plataforma nos permitiría poder gestionar nuestro sistema de riego desde el sistema domótico que se tenga en el hogar.

- **Integrar Panel Solar:** Una mejora al sistema sería integrar un panel solar que permita captar la radiación solar y alimentar a las minibombas de agua. Esto elevaría un poco el valor total del sistema, pero es una mejora a considerar.

4.3 Posibilidad de producto comercial

Aplicando las mejoras definidas en la sección anterior en nuestro prototipo, se podría obtener un producto comercial muy completo y a pesar de que son mejoras considerables, aún seguiría siendo un producto de bajo costo.

Tomando como referencia los costos especificados en la sección 3.5 Presupuesto y aplicando las mejoras para una posible versión comercial, el presupuesto final sería:

Descripción de Recursos	Cantidad	Precio Unitario (UF)	Valor Total (UF)
Caja para protección de intemperie	1	0,24 UF	0,24 UF
Batería 5V	1	0,11 UF	0,11 UF
Tarjeta Micro SD	1	0,12 UF	0,12 UF
Módulo Lector Memoria Micro SD	1	0,067 UF	0,067 UF
Panel Solar	1	0,20 UF	0,20 UF
TOTAL PRESUPUESTO MEJORAS			0,737 UF
TOTAL PRESUPUESTO SECCIÓN 3.5			2,117 UF
TOTAL FINAL DE PRESUPUESTO			2,854 UF

Los valores indicados son con IVA incluido.
El presupuesto final es de 2,854 UF que equivalen aproximadamente a $85.040 pesos chilenos. En base a este presupuesto, el costo del sistema sigue siendo inferior a los actuales sistemas de riego comerciales que aportan similares características.
Además, como se comentó anteriormente, los costos de los componentes al adquirirlos en grandes cantidades permitiría que el costo total baje aún más.

REFERENCIAS BIBLIOGRÁFICAS

[1] UAI, (2020, Jun 2),**"Escasez hídrica, la otra pandemia"**, [Online]. Disponible en: https://noticias.uai.cl/escasez-hidrica-la-otra-pandemia/.

[2] C. González, (2020, Sep 7),"Las pymes que promueven huertos a modo de terapia en el encierro", [Online]. Disponible en: https://www.paiscircular.cl/ciudad/las-pymes-que-promueven-huertos-a-modo-de-terapia-en-el-encierro/.

[3] AGROWARE, (2016, Nov 3),**"Tipos de riego y sus ventajas: ¿Cuál es el adecuado?"**, [Online]. Disponible en: https://sistemaagricola.com.mx/blog/tipos-de-riego-en-la-agricultura-y-ventajas/.

[4] Agriculturers, (2018, Jun 5),**"Qué es la agricultura inteligente"**, [Online]. Disponible en: https://agriculturers.com/que-es-la-agricultura-inteligente/.

[5] F. Guzmán, (2020, Nov 25),**"Agricultura inteligente: una revolución verde"**, [Online]. Disponible en: https://www.latercera.com/opinion/noticia/agricultura-inteligente-una-revolucion-verde/ZOK7HLF2IBGOTC24TZYRTQSHE4/.

[6] ODEPA, (2009, Jul),**"Agricultura de Precisión"**, [Online]. Disponible en: https://www.odepa.gob.cl/wp-content/uploads/2009/07/AgriculturaDePrecision.pdf.

[7] Y. González, (2020, Nov 25),**"Internet de las cosas: la nueva revolución verde"**, [Online]. Disponible en: https://www.bbva.com/es/internet-cosas-nueva-revolucion-verde/.

[8] S. Parent, (2021, Ene 7),**"¿Cómo influye la humedad en la calidad de los cultivos?"**, [Online]. Disponible en: https://www.pthorticulture.com/es/centro-de-formacion/como-influye-la-humedad-en-la-calidad-de-los-cultivos/.

[9] FAO, (2006, Ago),**"Evapotranspiración del cultivo"**, [Online]. Disponible en: http://www.fao.org/3/x0490s/x0490s.pdf.

[10] **"¿Qué es Arduino?"**, [Online]. Disponible en: https://arduino.cl/que-es-arduino/.

[11] E. Crespo, (2017, Jul 9),**"Aprendiendo Arduino"**, [Online]. Disponible en: https://aprendiendoarduino.wordpress.com/2017/07/09/i2c/.

[12] Robots Argentina, (2019, Jun 24),**"Descripción y funcionamiento del Bus I2C"**, [Online]. Disponible en: http://robots-argentina.com.ar/didactica/descripcion-y-funcionamiento-del-bus-i2c/.

[13] J. Mattassi, (2015, Oct 1),**"Innovación chilena permite tener el mejor jardín ahorrando un 50% de agua"**, [Online]. Disponible en: https://www.fayerwayer.com/2015/10/innovacion-chilena-permite-tener-el-mejor-jardin-ahorrando-un-50-de-agua/?fbclid=IwAR1qrQLReDJXUuhTZF9A5hJxAqz824boyfrs92UOBgEJtq34GpHuh4pdUzk.

[14] E-MICRONICA,**"Sensores, actuadores y elementos del sistema de control"**, [Online]. Disponible en: http://www.micronica.es/files/pdfs/SIHD/SIHD_Sens_Actu_EC.pdf.

[15] Naylamp Mechatronics,**"Módulo sensor de luz BH1750"**, [Online]. Disponible en: https://naylampmechatronics.com/blog/44_tutorial-modulo-sensor-de-luz-bh1750.html.

[16] **"Programa tu Arduino"**, [Online]. Disponible en: https://arduino.cl/programa-arduino/.

[17] S. Vihan, (2020, Dic 8),**"Connect Google Firebase with Nodemcu ESP8266"**, [Online]. Disponible en: https://senuravihanjayadeva.medium.com/connect-google-firebase-with-nodemcu-esp8266-7905d4b759f9.

[18] ElectroPeak, (2019, Abr 18),**"Connecting Arduino to Firebase to Send & Receive Data"**, [Online]. Disponible en: https://create.arduino.cc/projecthub/electropeak/connecting-arduino-to-firebase-to-send-receive-data-cd8805.

[19] Github,***"Fritzing Parts"***, [Online]. Disponible en: https://github.com/OgreTransporter/fritzing-parts-extra.

[20] P. A. Gaete,**"Gestión de recurso hídrico con tecnología IoT para el cultivo agrícola de la Región de la Araucanía"**, Magíster en Tecnologías de la Información, Departamento de Informática, Universidad Técnica Federico Santa María, Chile, 2011.

[21] D.F Apaza,**"Diseño e implementación de un sistema automatizado para riego tecnificado basado en el balance de humedad de suelo con tecnología Arduino en el laboratorio de control y automatización EPIME 2016"**, Tesis, Facultad de Ingeniería Mecánica Eléctrica, Electrónica y Sistemas, Universidad Nacional del Altiplano, Puno, Perú, 2017.

GLOSARIO

A continuación se detallan los conceptos mencionados en el trabajo y su significado:

AGRICULTURA DE AUTOCONSUMO: Consumo por parte de los productores de bienes o servicios de los productos que ellos mismos producen.
AUTÓMATA PROGRAMABLE: Es aquel artefacto o computador que está programado para automatizar una labor.
BIG DATA: Big data es un término que describe el gran volumen de datos estructurados y no estructurados los cuales precisan de aplicaciones informáticas no tradicionales de procesamiento de datos para tratarlos adecuadamente.
CAPACITANCIA: Es la capacidad de un componente o circuito para recoger y almacenar energía en forma de carga eléctrica.
COMANDOS AT: Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un dispositivo.
CONDUCTIVIDAD: Es la medida de la capacidad de un material o sustancia para dejar pasar la corriente eléctrica a través de él.
DPV (Déficit de Presión de Vapor): Diferencia entre la cantidad de vapor de agua que puede retener la atmósfera y la que contiene en ese momento.
ELECTRODOS: Un electrodo es un conductor eléctrico utilizado para hacer contacto con una parte no metálica de un circuito, por ejemplo un semiconductor, un electrolito, etc.
ELETROVÁLVULAS: Una electroválvula es una válvula electromecánica, diseñada para controlar el paso de un fluido por un conducto o tubería. La válvula se mueve mediante una bobina solenoide. Generalmente no tiene más que dos posiciones: abierto y cerrado, o todo y nada.
EROSIÓN DEL SUELO: La erosión del suelo es el desplazamiento de la capa superior de este, una forma de degradación del suelo. Un bajo nivel de erosión del suelo es un proceso natural en toda la tierra, pero las prácticas agrícolas pueden intensificar el proceso de erosión.
ESTOMAS: Los estomas son aperturas o poros microscópicos situados en la superficie de las hojas de las plantas a través de las cuales ocurre el intercambio gaseoso entre estas y el ambiente que las rodea. Son uno de los participantes en la fotosíntesis y también en la respiración de las plantas.
ESTRÉS HÍDRICO: El estrés por déficit hídrico o por sequía se produce en las plantas en respuesta a un ambiente escaso en agua, en donde la tasa de transpiración excede a la toma de agua.

FOTOSÍNTESIS: Es el proceso físico-químico por el cual las plantas, las algas y las bacterias fotosintéticas utilizan la energía de la luz solar para sintetizar compuestos orgánicos. Se trata de un proceso fundamental para la vida sobre la tierra y tiene un profundo impacto sobre la atmósfera y el clima terrestre.
GND: Concepto básico de electrónica, que indica el valor de referencia de tensiones en un circuito electrónico; normalmente todos los elementos del circuito se conectan a este punto en uno de sus terminales.
GNU: GNU es un sistema operativo de tipo Unix y está formado en su totalidad por software libre, mayoritariamente bajo términos de copyleft.
GRAVIMÉTRICA: El método Gravimétrico consiste en la medición de la aceleración de gravedad sobre un terreno con el fin de detectar las variaciones de densidades en las unidades geológicas presentes en el subsuelo.
HIGRÓMETRO: Sistema de medición de la humedad atmosférica.
HUELLA DE CARBONO: Huella de carbono es un indicador ambiental que pretende reflejar «la totalidad de gases de efecto invernadero (GEI) emitidos por efecto directo o indirecto de un individuo, organización, evento o producto».
I²C: Abreviatura de Inter-IC (Circuito inter-integrado). Es un bus serie de datos desarrollado en 1982 por Philips Semiconductors. Se utiliza principalmente internamente para la comunicación entre diferentes partes de un circuito, por ejemplo, entre un controlador y circuitos periféricos integrados.
IDE (Entorno de Desarrollo Integrado): Es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.
IoT (Internet de las cosas): Es un concepto que se refiere a una interconexión digital de objetos cotidianos con internet, utilizando software y sensores integrados para comunicarse entre sí.
INTELIGENCIA ARTIFICIAL (IA): Es la inteligencia llevada a cabo por máquinas.
ISLAS DE CALOR: Es una situación urbana, en donde se produce un aumento de la temperatura por la pavimentación y las edificaciones en las ciudades.
LUXÓMETRO: Sistema de medición de la iluminancia.
MAESTRO / ESCLAVO: Es un modelo de comunicación o control asimétrico donde un dispositivo o proceso (el "maestro") controla uno o más dispositivos o procesos (los "esclavos") y sirve como su centro de comunicación.
MICROCONTROLADOR: Es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales que cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida.

OPEN SOURCE: O también llamado software de código abierto, es el software cuyo código fuente y otros derechos que normalmente son exclusivos para quienes poseen los derechos de autor, son publicados bajo una licencia de código abierto o forman parte del dominio público. Frecuentemente el software de código abierto se desarrolla de manera colaborativa y los resultados se publican en internet.
PIVOT: Los sistemas de riego mediante pívot son sistemas de riego móviles que permiten regar grandes superficies.
PLÁNTULA: Es un término para nombrar a las primeras etapas de desarrollo de la planta, desde que germina la semilla hasta que adquiere sus primeras hojas verdaderas.
RED SMART AKIS: Es una red europea que difunde tecnologías y soluciones de Smart Farming (Agricultura Inteligente) entre los agricultores europeos y que acerca a profesionales de la agricultura, la industria y la investigación para la identificación y desarrollo conjunto de soluciones de Smart Farming en respuesta a las necesidades de los agricultores.
SEGURIDAD ALIMENTARIA: La seguridad alimentaria hace referencia a la disponibilidad suficiente y estable de alimentos, su acceso oportuno y su aprovechamiento biológico, de manera estable a través del tiempo.
SENSORES CAPACITIVOS: Son un tipo de sensor eléctrico, los cuales reaccionan ante metales y no metales que al aproximarse a la superficie activa sobrepasan una determinada capacidad. La distancia de conexión respecto a un determinado material es tanto mayor cuanto más elevada sea su constante dieléctrica.
SISTEMA EMBEBIDO: También conocido como “empotrado”, “incrustado” o “integrado”, es un sistema de computación diseñado para realizar funciones específicas, y cuyos componentes se encuentran integrados en una placa base “motherboard”.
SMART FARMING: El smart farming consiste en la utilización de las nuevas tecnologías (TIC) en la agricultura para aumentar la cantidad y la calidad de la producción, aprovechando al máximo los recursos y minimizando el impacto medioambiental.
SOBERANÍA ALIMENTARIA: La soberanía alimentaria es el derecho de los pueblos a definir sus propias políticas y estrategias sustentables de producción, distribución y consumo de alimentos con base en la pequeña y mediana producción.
SKETCH: Un sketch es una lista de ‘instrucciones’ para la placa Arduino. Los sketches se escriben usando el software de Arduino. Después de escribir un sketch, se puede cargar en una placa Arduino. Después de cargar un sketch a la placa Arduino, ésta comenzará a ejecutar las instrucciones que están escritas en el sketch.
TENSIOMÉTRICA: Las mediciones tensiométricas miden la cantidad de agua que requiere el cultivo. Los tensiómetros tradicionales, emulan la estructura y comportamiento de la raíz del cultivo, lo que permite que la humedad del suelo interactúe con el instrumento a través de la punta cerámica.

TERMISTOR: Es un dispositivo que permite detectar la temperatura a través de cambios de resistencia según el calor o frío detectado.
TIC: Las TIC o tecnologías de la información y la comunicación, son tecnologías que utilizan la informática, la microelectrónica y las telecomunicaciones para crear nuevas formas de comunicación a través de herramientas de carácter tecnológico y comunicacional, esto con el fin de facilitar la emisión, acceso y tratamiento de la información.
TRANSDUCTORES: Un transductor es un dispositivo capaz de transformar o convertir una determinada manifestación de energía de entrada, en otra diferente de salida, pero de valores muy pequeños en términos relativos con respecto a un generador.
UART: Transmisor-Receptor Asíncrono Universal, es el dispositivo que controla los puertos y dispositivos serie. Se encuentra integrado en la placa base o en la tarjeta adaptadora del dispositivo. Las funciones principales de este dispositivo es manejar las interrupciones de los dispositivos conectados al puerto serie y convertir los datos en formato paralelo, transmitidos al bus de sistema, a datos en formato serie, para que puedan ser transmitidos a través de los puertos y viceversa.
VCC: Es el voltaje de suministro de un circuito eléctrico.

ANEXOS

1. Programa placa Arduino

```
#include <ArduinoJson.h>
#include <SoftwareSerial.h>
#include <Wire.h>
#include <BH1750.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
#include <DHT_U.h>

#define soilWet 220   // Define max value we consider soil 'wet'
#define soilDry 526   // Define min value we consider soil 'dry'
#define soil_sensor A0 // Capacitive Soil Moisture Sensor
#define LCD_ADDR 0x27 // Setting i2c LCD address
#define BH1750_ADDR 0x23 // Setting i2c BH1750 address
#define DHTPIN 7     // Sensor DHT connected to pin 7
#define DHTTYPE DHT22   // Sensor DHT 22  (AM2302)

DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(LCD_ADDR,20,4);
BH1750 luxSensor;
SoftwareSerial ArduinoUno(3,2);

int temp, hum, soilMoisture, percentage;
int relay = 8;
long nightLight = 20;
long dayLight = 30;
String stateLight = "";
String stateWater = “";

void setup() {

 Serial.begin(9600);
 ArduinoUno.begin(4800);
 dht.begin();
 Wire.begin();
 luxSensor.configure(BH1750::ONE_TIME_HIGH_RES_MODE);
```

```
  luxSensor.begin(BH1750_ADDR);

  if (luxSensor.begin()) {
    Serial.println(F("Inicializando sensor BH1750..."));
  }
  else {
    Serial.println(F("Error inicializando sensor BH1750"));
  }

  lcd.init();
  lcd.backlight();
  lcd.clear();
  lcd.setCursor(0, 0);

  pinMode(relay, OUTPUT);
  pinMode(soil_sensor, INPUT);
  digitalWrite(relay, HIGH); //turn off relay

  delay(500);
}

StaticJsonBuffer<200> jsonBuffer;
JsonObject& doc = jsonBuffer.createObject();

void loop() {

  hum = dht.readHumidity();
  temp = dht.readTemperature();

  uint16_t lux = luxSensor.readLightLevel();

  soilMoisture = analogRead(soil_sensor);
  percentage = map(soilMoisture, soilWet, soilDry, 100, 0);

  //Avoid negative percentages
  if(percentage < 0) percentage = 0;
  if(percentage > 100) percentage = 100;

  if (isnan(hum) || isnan(temp)) {
    Serial.println("Falló la lectura del sensor DHT22!");
    return;
  }
```

```
else
{
 Serial.print("{\"Humedad\":");
 Serial.print(hum);
 Serial.print("% ,");
 Serial.print("\"Temperatura\":");
 Serial.print(temp);
 Serial.print("ºC ,");
 Serial.print("\"Humedad del Suelo\":");
 Serial.print(percentage);
 Serial.println("% ,");
 Serial.print("\"Iluminancia\":");
 Serial.print(lux);
 Serial.println("lx }");

 if(lux<nightLight) {
  Serial.println("Es de Noche\n");
  stateLight = "Noche";
 }
 else if(lux>dayLight) {
   Serial.println("Es de Dia\n”);
   stateLight = "Dia";
 }

 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("Temp:");
 lcd.print(temp);
 lcd.print((char)223);
 lcd.setCursor(10,0);
 lcd.print("Hum:");
 lcd.print(hum);
 lcd.print("%");
 lcd.setCursor(0,1);
 lcd.print("Hum Suelo:");
 lcd.print(percentage);
 lcd.print("%");
 lcd.setCursor(0,2);
 lcd.print("Iluminancia:");
 lcd.print(lux);
 lcd.print(" lx");
```

```
    if(lux<nightLight) {
      lcd.setCursor(0,3);
      lcd.print("Es de Noche!");
    }
    else if(lux>dayLight) {
      lcd.setCursor(0,3);
      lcd.print("Es de Dia!");
    }
  }

// Determine status of our soil sensor
  if (soilMoisture < soilWet) {
    Serial.println("Status: El suelo está demasiado húmedo");
    stateWater = "False";
    digitalWrite(relay, HIGH); // turn off pump 5 seconds
    delay(5000);
    Serial.println("Relay apagado\n");
  } else if (soilMoisture >= soilWet && soilMoisture < soilDry) {
    Serial.println("Status: La humedad del suelo es perfecta");
    stateWater = "False";
    digitalWrite(relay, HIGH); // turn off pump 5 seconds
    delay(5000);
    Serial.println("Relay apagado\n");
  } else {
    Serial.println("Status: El suelo está demasiado seco, ¡Es tiempo de regar!");
    stateWater = "True";
    digitalWrite(relay, LOW); // turn on pump 5 seconds
    delay(5000);
    Serial.println("Relay encendido\n");
  }

  doc["temp"] = temp;
  doc["hum"] = hum;
  doc["soilMoisture"] = percentage;
  doc["state_light"] = stateLight;
  doc["state_water"] = stateWater;
  doc["message"] = "1523";

  if(ArduinoUno.available()>0)
  {
   doc.printTo(ArduinoUno);
  }
```

```
  Serial.println("Datos enviados a Módulo WiFi");
  doc.prettyPrintTo(Serial);
  Serial.println("\n");

  delay(1000);
}
```

2. Programa placa NodeMCU

```
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
#include <SoftwareSerial.h>
#include <ArduinoJson.h>

#define FIREBASE_HOST "arduino-nodemcu-project-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "vaztQo1fXD1a9wEJ4CgnChYKiUeXFVoHBZXcnrhH"
#define WIFI_SSID "VTR-7364922"
#define WIFI_PASSWORD "q7vyGmnKnqdp"

SoftwareSerial NodeMCU(D2,D1);

void setup() {

  Serial.begin(9600);
  NodeMCU.begin(4800);
  while (!Serial) continue;

  // connect to wifi.
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Conectandose al WiFi");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("Conectado exitosamente: ");
  Serial.println(WiFi.localIP());

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
```

```
 if(Firebase.failed())
 {
  Serial.print(Firebase.error());
 }
 else{
  Serial.print("Firebase conectado exitosamente\n");
 }

 pinMode(D2, INPUT);
 pinMode(D1, OUTPUT);

 delay(500);
}

void loop() {

   StaticJsonBuffer<200> jsonBuffer;
   JsonObject& doc = jsonBuffer.parseObject(NodeMCU);

   if (doc == JsonObject::invalid())
   {
     jsonBuffer.clear();
     return;
  }

  //Print the data in the serial monitor
 Serial.println("Datos recibidos desde Arduino");
 doc.prettyPrintTo(Serial);
 Serial.println("");
 Serial.print("Temperatura ");
 int data1 = doc["temp"];
 Serial.println(data1);
 Serial.print("Humedad    ");
 int data2 = doc["hum"];
 Serial.println(data2);
 Serial.print("Humedad de suelo    ");
 int data3 = doc["soilMoisture"];
 Serial.println(data3);
 Serial.print("Iluminancia    ");
 String data4 = doc["state_light"];
 Serial.println(data4);
```

```
 Serial.print("Estado del Riego    ");
 String data5 = doc["state_water"];
 Serial.println(data5);
 Serial.print("Número Mensaje    ");
 String data6 = doc["message"];
 Serial.println(data6);
 Serial.println("");

// set value
 Firebase.setInt("/Planta1/Temperatura", data1);
 // handle error
 if (Firebase.failed()) {
    Serial.print("La configuración de /Planta1/Temperatura falló:");
    Serial.print(Firebase.error());
    return;
 }
 delay(1000);

 // set value
 Firebase.setInt("/Planta1/Humedad", data2);
 // handle error
 if (Firebase.failed()) {
    Serial.print("La configuración de /Planta1/Humedad falló:");
    Serial.println(Firebase.error());
    return;
 }
 delay(1000);

 // set value
 Firebase.setInt("/Planta1/Humedad_Suelo", data3);
 // handle error
 if (Firebase.failed()) {
    Serial.print("La configuración de /Planta1/Humedad_Suelo falló:");
    Serial.println(Firebase.error());
    return;
 }
 delay(1000);

 // set value
 Firebase.setString("/Planta1/Status_Iluminancia", data4);
 // handle error
 if (Firebase.failed()) {
```

```
    Serial.print("La configuración de /Planta1/Status_Iluminancia falló:");
    Serial.println(Firebase.error());
    return;
  }
  delay(1000);

  // set value
  Firebase.setString("/Planta1/Status_Riego", data5);
  // handle error
  if (Firebase.failed()) {
    Serial.print("La configuración de /Planta1/Status_Riego falló:");
    Serial.println(Firebase.error());
    return;
  }
  delay(1000);

  // set value
  Firebase.setString("/Planta1/Mensaje", data6);
  // handle error
  if (Firebase.failed()) {
    Serial.print("La configuración de /Planta1/Mensaje falló:");
    Serial.println(Firebase.error());
    return;
  }
  delay(1000);
}
```

3. Calibración Sensor de Humedad

```
#define hum_suelo A0
int humedad_suelo;

void setup() {

  Serial.begin(9600);

}

void loop() {

  humedad_suelo = analogRead(hum_suelo);
  Serial.print("Valor del sensor de humedad capacitivo: ");
  Serial.println(humedad_suelo);
  delay(1000);

}
```