

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

APLICACIONES CON SISTEMAS EMBEBIDOS

**IMPLEMENTACIÓN DE UN PROTOTIPO PARA LA DETECCIÓN
DEL NIVEL DE HUMEDAD DEL SUELO E ILUMINACIÓN DE UN
JARDÍN UTILIZANDO SISTEMAS EMBEBIDOS**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
TELECOMUNICACIONES**

AXEL RAÚL CUAMACAS ROSERO
axel.cuamacas@epn.edu.ec

DIRECTOR: Dra. SORAYA LUCÍA SINCHE MAITA
soraya.sinche@epn.edu.ec

Quito, abril de 2023

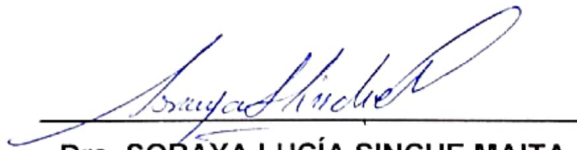
CERTIFICACIONES

Yo, Axel Raúl Cuamacás Rosero declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



AXEL RAÚL CUAMACÁS ROSERO

Certifico que el presente trabajo de integración curricular fue desarrollado por AXEL RAÚL CUAMACÁS ROSERO, bajo mi supervisión.



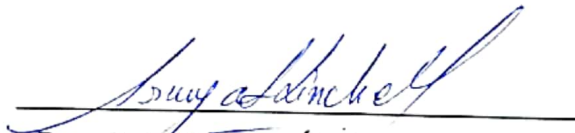
Dra. SORAYA LUCÍA SINCHE MAITA

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.



AXEL RAÚL CUAMACÁS ROSERO



Dra. SORAYA LUCÍA SINCHE MAITA

DEDICATORIA

El presente trabajo se lo dedico a mi amada madre Patricia, quien ha sido mi principal apoyo a lo largo de mi etapa universitaria y quien me ha inspirado con su fuerza, dedicación y amor incondicional. Gracias por tu paciencia y por creer en mí siempre, incluso cuando yo mismo no lo hacía. Todo lo que he conseguido es gracias a tu amor y sacrificio.

AGRADECIMIENTO

A mi madre: Patricia Rosero

Por su amor, apoyo y dedicación total hacia mi durante toda mi etapa universitaria, sin tu apoyo nada de esto habría sido posible.

A mi padre: Raúl Cuamacás

Por tu motivación y consejos durante mi etapa universitaria, tus palabras de aliento han sido primordiales para mis logros académicos y personales.

A mis abuelos: José Rosero y Aura Ayala

Por su amor y motivación constante durante toda mi vida, pues sus consejos y ejemplo de perseverancia me han permitido alcanzar este logro.

A mi tía: Betty Rosero

Por su apoyo y motivación brindado en mis primeros años universitarios, su confianza y palabras de aliento siempre me ayudaron a superarme.

A Soraya L. Sinche Maita, PhD

Por su guía y tiempo dedicado en la dirección de este trabajo, su conocimiento y experiencia han sido fundamentales para el desarrollo del mismo.

A la Escuela Politécnica Nacional

Por brindarme una educación de calidad, pues los conocimientos y habilidades adquiridas en todo este tiempo me han preparado para ser un gran profesional.

ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE TABLAS	X
ÍNDICE DE ECUACIONES.....	XI
ÍNDICE DE CÓDIGOS	XII
RESUMEN	XIII
ABSTRACT	XIV
1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO	1
1.1 Objetivo general	2
1.2 Objetivos específicos	2
1.3 Alcance	2
1.4 Marco Teórico	2
1.4.1 Humedad en el Suelo e Iluminación para el Cuidado a las Plantas.....	3
1.4.1.1 Humedad en el Suelo para las Plantas	3
1.4.1.2 Iluminación para las Plantas.....	6
1.4.2 Terapia Hortícola.....	8
1.4.3 Internet de las Cosas	9
1.4.3.1 Automatización a la Agricultura mediante Sistemas IoT	9
1.4.3.2 Sistemas Embebidos.....	9
1.4.3.3 Sensores utilizados en la Agricultura	10
1.4.3.4 Sensores de Humedad en el Suelo	10
1.4.3.5 Sensores de Luz ambiental.....	11
1.4.4 Plataformas para Monitoreo de Datos utilizando Sistemas Embebidos.....	12
2 METODOLOGÍA	13
2.1 Descripción de la Metodología utilizada	13
2.2 Requerimientos de Hardware para el Prototipo	14
2.2.1 Módulo Wi-Fi ESP32.....	14
2.2.2 Sensor Capacitivo de Humedad en el Suelo v1.2	16
2.2.3 Sensor de Luz digital BH1750	16

2.2.4	Relé HK19F DC5V-SHG.....	17
2.2.5	Mini bomba de agua sumergible 3-6V	18
2.3	Requerimientos de Software para el Prototipo.....	18
2.3.1	Arduino IoT Cloud.....	18
2.3.2	IFTTT	18
2.3.3	EasyEDA.....	19
2.4	Diseño del Prototipo.....	20
2.4.1	Diseño del Circuito Esquemático.....	21
2.4.2	Comunicación del ESP32 con la plataforma de Arduino Cloud IoT.....	21
2.4.3	Comunicación del Hidrómetro con el ESP32.....	23
2.4.4	Comunicación del Sensor de Luz digital BH1750 con el ESP32	24
2.4.5	Activación de Bomba de Agua para Suelos Secos	25
2.4.6	Activación de Alerta en Baja Iluminancia del Jardín	30
2.4.7	Alimentación del Circuito	31
2.4.8	Diagrama de Flujo	32
2.5	Implementación del prototipo	33
2.5.1	Desarrollo de la PCB	34
2.5.2	Desarrollo del Módulo Protector	37
3	PRUEBAS, RESULTADOS, CONCLUSIONES Y RECOMENDACIONES....	39
3.1	Pruebas de Funcionamiento	39
3.1.1	Prueba de los Sensores	39
3.1.2	Recopilación de Información en una Base de Datos para Diferentes Ambientes de Prueba.....	39
3.1.3	Comparación entre los Valores obtenidos por el Sensor BH1750 y un luxómetro..	41
3.1.4	Nivel de Iluminancia con el Sensor de Luz Digital BH1750	42
3.1.5	Nivel de Humedad en el Suelo obtenido con el Sensor Capacitivo de Humedad en el Suelo v1.2.....	43
3.1.6	Prueba de Funcionamiento del envío de correos de alerta	43
3.1.7	Prueba de Funcionamiento de la bomba de agua	44
3.1.8	Prototipo en Ambientes de Prueba Outdoor e Indoor	44
3.1.9	Pruebas de Visualización y Monitoreo de datos en Arduino IoT Cloud.....	46
3.2	Resultados	47
3.3	Conclusiones.....	51
3.4	Recomendaciones.....	52
4	REFERENCIAS BIBLIOGRÁFICAS	54

5 ANEXOS

ANEXO I. Código Fuente Principal

ANEXO II. Códigos Generador de Base de Datos Para Pruebas en Diferentes Ambientes

ANEXO III. Manual de Conexión del ESP32 con la Plataforma de Arduino Cloud IoT

ANEXO IV. Envío de alerta por email a través de la plataforma IFTTT

ANEXO V. Enlace de Pruebas de Funcionamiento

ÍNDICE DE FIGURAS

Figura 2.1 Metodología cascada	13
Figura 2.2 Módulo Wi-Fi ESP32 y su distribución de pines.....	15
Figura 2.3 Sensor capacitivo de humedad en el suelo v1.2	16
Figura 2.4 Sensor de Luz digital BH1750	17
Figura 2.5 Relé HK19F DC5V-SHG	17
Figura 2.6 Mini bomba de agua sumergible 3-6V	18
Figura 2.7 Ejemplo para la arquitectura de IFTTT	20
Figura 2.8 Diagrama de funcionamiento del prototipo.....	21
Figura 2.9 Circuito Esquemático del prototipo.....	22
Figura 2.10 Modo de conexión del sensor de humedad en el suelo v1.2 con el ESP32	23
Figura 2.11 Modo de conexión del sensor de luz BH1750 con el ESP32	25
Figura 2.12 Relé controlado por transistor con niveles de tensión TTL, CMOS, LVTTTL y LVCMOS.....	26
Figura 2.13 Transistor bipolar NPN 2N2222A	27
Figura 2.14 Diodo rectificador 1N4007	27
Figura 2.15 Resistencia de 10 K Ω	28
Figura 2.16 Distribución de pines del Relé HK19F visto desde la parte inferior... ..	28
Figura 2.17 Modo de conexión de la bomba de agua con el ESP32.....	29
Figura 2.18 Cargador que alimenta el ESP32	31
Figura 2.19 Pila recargable que alimenta la bomba de agua	32
Figura 2.20 Diagrama de flujo para la detección del nivel de humedad del suelo e iluminación de un jardín	33
Figura 2.21 Distribución de los componentes electrónicos en la PCB	34
Figura 2.22 Visualización en 3D de la PCB en su capa inferior y superior.....	34
Figura 2.23 Transferencia térmica del circuito a la placa de fibra de vidrio FR4 ..	35
Figura 2.24 Proceso de quemado de la PCB	35
Figura 2.25 Protección del circuito impreso y perforación de los agujeros diseñados.....	36
Figura 2.26 Soldado de los elementos en la PCB.....	36
Figura 2.27 Resultado final de la PCB diseñada.....	37
Figura 2.28 Diseño del módulo protector	37
Figura 2.29 Prototipo final protegido por el modulo protector.....	38
Figura 3.1 Envío de datos a MySQL desde el ESP32 utilizando el método HTTP POST	40
Figura 3.2 Base de datos en phpMyAdmin	40
Figura 3.3 Grafica de la Iluminancia de la aplicación y del sensor BH1750	42
Figura 3.4 Alerta generada vía Email	43
Figura 3.5 Activación automática de la bomba de agua	44
Figura 3.6 Pruebas en un ambiente Outdoor	45
Figura 3.7 Pruebas en un ambiente Indoor	45
Figura 3.8 Pruebas en un ambiente Indoor con luz artificial.....	46
Figura 3.9 Visualización y Monitoreo de datos en Arduino IoT Cloud mediante la computadora	46

Figura 3.10 Visualización y Monitoreo de datos en Arduino IoT Cloud mediante el celular.....	47
Figura 3.11 Gráfica de Iluminancia y Humedad en el Escenario 1.....	48
Figura 3.12 Gráfica de Iluminancia y Humedad en el Escenario 2.....	49
Figura 3.13 Gráfica de Iluminancia y Humedad en el Escenario 3.....	50
Figura 3.14 Gráfica de Iluminancia y Humedad en el Escenario 4.....	51

ÍNDICE DE TABLAS

Tabla 1.1 Métodos para medir la humedad del suelo.....	4
Tabla 1.2 Condiciones para determinar la cantidad de agua en el suelo	4
Tabla 1.3 Valores de densidad aparente para diferentes texturas de suelo	5
Tabla 1.4 Niveles de humedad en el suelo recomendados para la mayoría de plantas [11].....	6
Tabla 1.5 Valores de iluminación recomendado para cierto tipo de cultivos	7
Tabla 1.6 Efectos de la Iluminación en las plantas	7
Tabla 1.7 Influencia de distintos rangos de iluminancia en las plantas	8
Tabla 1.8 Tipos de sensores y su descripción.....	10
Tabla 1.9 Tipos de sensores para medir humedad del suelo	11
Tabla 1.10 Funcionamiento de los diferentes tipos de sensores de luz ambiental	12
Tabla 1.11 Comparación de plataformas para el desarrollo de proyectos IoT	12
Tabla 2.1 Comparación entre las características de Arduino Uno, ESP8266 y ESP32	15
Tabla 2.2 Características de funcionamiento del Relé HK19F DC5V-SHG.....	17
Tabla 2.3 Características de Arduino IoT Cloud	19
Tabla 2.4 Pines analógicos y digitales del ESP32 utilizados para el prototipo	21
Tabla 2.5 Conexión de los pines del ESP32 con los pies del sensor capacitivo de humedad en el suelo v1.2	23
Tabla 2.6 Conexión de los pines del ESP32 con los pies del sensor de luz BH1750	25
Tabla 3.1 Valores de Iluminancia de la aplicación y del sensor BH1750	42
Tabla 3.2 Muestras de Iluminancia y Humedad en el Escenario 1	47
Tabla 3.3 Muestras de Iluminancia y Humedad en el Escenario 2.....	48
Tabla 3.4 Muestras de Iluminancia y Humedad en el Escenario 3.....	49
Tabla 3.5 Muestras de Iluminancia y Humedad en el Escenario 4.....	50

ÍNDICE DE ECUACIONES

Ecuación 1.1 Contenido de agua gravimétrico	5
Ecuación 1.2 Densidad aparente	5
Ecuación 1.3 Contenido volumétrico del agua	5
Ecuación 2.1 Resistencia en base del transistor para activar la bomba de agua.	27
Ecuación 3.1 Error de iluminancia (%)	41

ÍNDICE DE CÓDIGOS

Código 2.1 Conexión del ESP32 con Arduino IoT Cloud	22
Código 2.2 Sección del código para la obtención del % de humedad en las plantas	24
Código 2.3 Sección del código necesaria para la obtención la iluminancia en luxes	25
Código 2.4 Sección del código para la activación de la bomba de agua en suelos secos	29
Código 2.5 Sección del código para enviar un correo de alerta al detectar niveles bajos de iluminancia en el jardín	30

RESUMEN

Hoy en día, muchas de las tareas relacionadas con la jardinería dependen del trabajo manual, lo que puede resultar en un aumento de los costos y el tiempo requerido para mantener y cuidar un jardín. Por esta razón, es importante buscar formas de mejorar la eficiencia, la productividad y la sostenibilidad de la jardinería.

En el presente trabajo se presenta la implementación de un prototipo para la detección del nivel de humedad del suelo e iluminación de un jardín a través de sistemas embebidos. El prototipo implementado aumenta la eficiencia de las tareas relacionadas con la jardinería al reducir el trabajo manual y el tiempo requerido para el cuidado y mantenimiento de un jardín. Al automatizar los procesos relacionados con la jardinería se logra una mejora en la siembra, riego y mantenimiento de las plantas.

El trabajo de integración curricular comprende tres capítulos. En el capítulo 1 se realiza la descripción del componente y se exponen los objetivos a cumplir. Se realiza un análisis de los requisitos necesarios, así como de los conocimientos relacionados con la automatización de la jardinería. En el capítulo 2 se especifica cada elemento a utilizar y la forma en que se incorporarán al prototipo. En el capítulo 3 se presentan las pruebas del prototipo realizadas en diferentes ambientes para que en base a los resultados se generen las correspondientes conclusiones y recomendaciones. Finalmente, se presenta la bibliografía utilizada y los anexos que se consideran de utilidad para el trabajo.

PALABRAS CLAVE: Jardinería, automatización, sistemas embebidos, IoT.

ABSTRACT

Today, many of the tasks related to gardening depend on manual labor, which can result in increased costs and the time required to maintain and care for a garden. For this reason, it is important to look for ways to improve the efficiency, productivity, and sustainability of gardening.

In the present work, the implementation of a prototype for the detection of the level of soil humidity and lighting of a garden through embedded systems is presented. The implemented prototype increases the efficiency of tasks related to gardening by reducing manual labor and the time required for the care and maintenance of a garden. By automating the processes related to gardening, an improvement is achieved in planting, watering and maintenance of plants.

The work of curricular integration comprises three chapters. In Chapter 1 the description of the component is made and the objectives to be met are exposed. An analysis of the necessary requirements is carried out, as well as the knowledge related to the automation of gardening. Chapter 2 specifies each element to be used and the way in which they will be incorporated into the prototype. Chapter 3 presents the prototype tests carried out in different environments so that the corresponding conclusions and recommendations are generated based on the results. Finally, the bibliography used and the annexes that are considered useful for the work are presented.

KEYWORDS: *Gardening, automation, embedded system, IoT.*

1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

En la actualidad, los sistemas embebidos son una herramienta computacional utilizada para ejecutar tareas de control en tiempo real y cubrir una necesidad específica al realizar una o varias funciones dedicadas [1]. Se considera que la automatización en la agricultura es de vital importancia dado que se mejoran los procesos realizados tradicionalmente, ayudando a elevar los estándares de crecimiento y cuidado que puede tener un jardín. Haciendo uso de los dispositivos mencionados se puede monitorear un jardín dentro de casa adecuadamente e incluso aumentar la extensión del jardín, contribuyendo a la estética de un hogar y aumentando el valor de plusvalía de la propiedad; además de que las actividades relacionadas con la jardinería brindan bienestar a las personas que sufren de estrés como es el caso de la terapia hortícola [2].

Con la ayuda de este tipo de dispositivos tales como sensores, microcontroladores y actuadores, a más de evaluar el estado del suelo y el nivel de iluminación al que se encuentra expuesto el jardín, se facilita el proceso de cuidado y mantenimiento del mismo. Por otra parte, al generar un historial detallado del comportamiento de las variables introducidas al sistema (humedad de la tierra y luminosidad), se permite tomar acciones a favor de las plantas sembradas en el jardín [3]. Como principales elementos a ser utilizados para el desarrollo del prototipo se tiene un microcontrolador con conectividad Wi-Fi, un sensor capacitivo de humedad y un sensor de luz digital. A través de una plataforma para proyectos IoT, se observan los niveles de humedad en el suelo y los niveles de iluminación.

En este contexto, el presente Trabajo de Integración Curricular busca dar una solución al cuidado de las plantas sembradas en jardines residenciales monitoreando la humedad del suelo y la iluminación en el ambiente, dos parámetros primordiales para el buen crecimiento de las plantas; además se genera un sistema de alerta que advierte al responsable del cuidado del jardín a través de mensajes enviados a su correo electrónico, también se tiene una bomba de agua que se enciende automáticamente cuando la planta lo requiera, esto se activa al momento en el que los sensores superan cierto umbral establecido en los niveles de humedad en el suelo e iluminación en el ambiente donde se encuentre el jardín.

1.1 Objetivo general

Implementar un prototipo para la detección del nivel de humedad del suelo e iluminación de un jardín utilizando sistemas embebidos.

1.2 Objetivos específicos

1. Presentar los conceptos de sistemas embebidos para la implementación del prototipo.
2. Determinar los niveles de humedad e iluminación recomendables a los que las plantas de un jardín pueden estar expuestas.
3. Desarrollar un prototipo de prueba para detección del nivel de humedad del suelo e iluminación.
4. Analizar los resultados obtenidos de las pruebas de funcionamiento del prototipo.

1.3 Alcance

Se presenta un estudio de los conceptos de sistemas embebidos y sensores que detecten el nivel de humedad en el suelo e iluminación de plantas en un jardín.

Se definen los componentes a ser utilizados en el prototipo de prueba, enfocándose en los requerimientos necesarios para la detección del nivel de humedad del suelo e iluminación de un jardín. Se implementa un prototipo de prueba en base a los requerimientos analizados anteriormente y se realizará las pruebas de funcionamiento del mismo. En caso de existir errores se realizan los ajustes que sean necesarios.

Se analizan los resultados obtenidos en diferentes ambientes de prueba, tales como días soleados, lluviosos y nublados, generando una base de datos de los valores sensados.

1.4 Marco Teórico

En la actualidad la mayoría de procesos se han automatizado, incluyendo procesos agrícolas, donde se reduce tiempo, recurso y errores humanos en beneficio de las plantas. Los sectores agrícolas se han visto afectados en su producción, por lo que es necesario buscar soluciones eficientes y económicas que ayuden a fortalecer el cuidado de las plantas, revitalizando este sector productivo [4]. En la [Figura 1.1](#) se presenta el estado de una planta saludable y una planta marchita debido a niveles de humedad en el suelo e iluminación inadecuados.



Figura 1.1 Planta Marchita vs Planta Saludable

1.4.1 Humedad en el Suelo e Iluminación para el Cuidado a las Plantas

La variación de la humedad en el suelo e iluminación afectan de forma positiva o negativa a los cultivos, dependiendo de la intensidad con la que es aplicada a las plantas y cuáles son los niveles adecuados para cada cultivo.

De la cantidad de humedad en el suelo dependerá qué tantos nutrientes pueden absorber las plantas, mientras que la luz proveerá de la energía necesaria para que las plantas efectúen la fotosíntesis. Es por esto que los conceptos desarrollados a continuación son de vital importancia para tener un cultivo saludable [5][6].

1.4.1.1 Humedad en el Suelo para las Plantas

La humedad en la tierra depende de varios factores como la lluvia, la cantidad de agua absorbida por cada tipo planta y la temperatura a la que está expuesta la planta. Siendo muy importante que la humedad en el suelo sea correcta para cada tipo de planta, para que los cultivos se mantengan en buen estado, dado que si se tiene un nivel de humedad inadecuado en el suelo las plantas no se desarrollan adecuadamente en su tamaño, floración y estado de los frutos [7].

La cantidad de agua en el suelo se puede determinar relacionando el peso del suelo seco con la cantidad de agua. La humedad en el suelo se puede expresar en volumen, peso o porcentaje, en la [Tabla 1.1](#) se presenta los métodos para medir la humedad del suelo [7].

Tabla 1.1 Métodos para medir la humedad del suelo [7][8]

Método	Descripción
Gravimétrico	Es la forma más usada para mostrar el contenido hídrico del suelo, midiendo directamente la humedad del suelo y comparando el contenido de agua obtenido en una muestra por el porcentaje del suelo seco en peso. Se usa la Ecuación 1.1 .
Volumétrico	Mide la humedad del suelo al multiplicar el contenido gravimétrico de agua por la densidad aparente (peso del suelo seco por unidad volumétrica del suelo). Se usa la Ecuación 1.2 y Ecuación 1.3
Eléctrico	Mide el valor de la resistencia, la inductancia, la conductividad y la capacidad del suelo.
Radiactivo	Mide las sustancias radioactivas del suelo.
Óptica	Mide la humedad dependiendo de la reflectividad de los flujos de luz.
Tensométrico	Mide la humedad en el suelo en unidades de presión negativa, siendo esta una medida de fuerza ejercida por las raíces de una planta al extraer agua de los poros del suelo.
Métodos exprés	Usando propiedades organolépticas (características físicas que tiene la materia y pueden percibirse de forma directa por los sentidos)

El suelo alberga agua en los micro y macro poros incorporados en partículas del suelo, donde para determinar el contenido hídrico del suelo se debe valorar las condiciones de la [Tabla 1.2](#) [9].

Tabla 1.2 Condiciones para determinar la cantidad de agua en el suelo [9]

Condición	Detalle
Humedad de saturación	Concentración de agua en los poros del suelo que están llenos de agua (se observa en riegos y lluvias abundantes).
Capacidad de campo (CC)	Límite máximo de agua en el suelo 2 o 3 días posteriores a un riego o lluvia intensa.
Punto de marchitez permanente (PMP)	Marchitez irreversible, donde la planta no puede absorber la cantidad de agua necesaria para sobrevivir
Agua útil (AU)	Diferencia entre capacidad de campo y punto de marchitez permanente. Porción de agua útil en la zona de crecimiento radical de las plantas.

Para calcular el nivel de humedad del suelo por el método gravimétrico y volumétrico se tiene la [Ecuación 1.1](#), [Ecuación 1.2](#) y [Ecuación 1.3](#).

$$C_g(\%) = \frac{P_h(g) - P_s(g)}{P_s(g)} \times 100 \quad \text{Ecuación 1.1}$$

$$D_{ap} = \frac{P_s(g)}{V_s} \quad \text{Ecuación 1.2}$$

$$C_v(\%) = \frac{D_{ap}}{D_a} \times C_g(\%) \quad \text{Ecuación 1.3}$$

Donde:

C_g : Contenido de agua gravimétrico (%)

P_h : Peso del suelo húmedo (g)

P_s : Peso del suelo seco (g)

D_{ap} : Densidad aparente (mg/m³)

V_s : Volumen del suelo

C_v : Contenido volumétrico del agua (%)

D_a : Densidad del agua

La textura del suelo es un parámetro relevante para determinar el tipo de suelo en el que se está cultivando y se lo determina por medio de la densidad aparente utilizando la [Ecuación 1.2](#). Los valores de densidad aparente para texturas de suelo arenosa, limosa y arcillosa se encuentran en la [Tabla 1.3](#) y se observa estas texturas en la [Figura 1.2](#).

Tabla 1.3 Valores de densidad aparente para diferentes texturas de suelo [9]

Textura del suelo	Densidad aparente (mg/m ³)
Arenosa	1,40 – 1,80
Limosa	1,25 – 1,40
Arcillosa	1,00 – 1,25



Figura 1.2 Textura de suelo Arenoso vs Limoso vs Arcilloso [10].

Los niveles de humedad en el suelo recomendados para las especies más comunes de plantas se encuentran especificados en la [Tabla 1.4](#). Es importante señalar que estos valores representan las especies más comunes, si se desea obtener un detalle más específico del tipo de humedad en el suelo requerido para una planta se debe acudir con un especializado en la horticultura [11].

Tabla 1.4 Niveles de humedad en el suelo recomendados para la mayoría de plantas [11]

Tipo de planta	Humedad en el suelo (%)
Flores	21 % - 40 %
Arboles	21 % - 40 %
Arbustos	21 % - 40 %
Vegetales	41 % - 80 %
Frutas	21 % - 60%

1.4.1.2 Iluminación para las Plantas

La fuente de luz y energía más fuerte e inagotable que existe en la tierra es la producida por la radiación solar, donde el sol emite longitudes de onda entre 280 y 2800 nm. Siendo la luz parte fundamental para el crecimiento adecuado de las plantas, se consideran varios factores importantes como la intensidad de la luz medida por fotones, o dependiendo la energía irradiada, su duración, la composición espectral y la dirección medida según el grado de dispersión y la ubicación de la fuente. La luz ayuda a que las plantas realicen el proceso de fotosíntesis [12].

El monitorear la intensidad de luz a la que están expuestas las plantas es muy importante para el cultivo, dado que cada tipo de planta requiere de niveles distintos de intensidad de luz, donde 100.000 lux es el equivalente a la luz producida por el sol en un día despejado.

En el caso de plantas como la lechuga y el jitomate requieren de 10.000 a 40.000 lux, por lo que es recomendable cubrir a las plantas con una malla sombra (malla color negro) o un plástico lechoso (plástico que distribuye la luz solar de igual forma a todas las plantas) y así reducir la intensidad de luz a la que están expuestas este tipo de plantas. En caso de que los cultivos se encuentren en lugares oscuros o no reciban la intensidad de luz adecuada, se requiere de luz artificial, que puede ser generada por focos de 200 a 400 w, ayudando a mejorar la luminosidad a la que están expuestas las plantas en base a las necesidades de cada tipo de planta [13].

En la [Tabla 1.5](#) se encuentran los valores de iluminación recomendado para varios tipos de plantas cultivadas comúnmente en jardines.

Tabla 1.5 Valores de iluminación recomendado para cierto tipo de cultivos [13]

Tipo de planta	Intensidad de luz sobre la superficie de la planta (lux)
Rosa	100000
Clavel	15000-45000
Crisantemo	75000-95000
Tomate	10000-40000
Lechuga	12000-30000

El indicador más apropiado para identificar si los niveles de luz son los correctos, es observar el estado en el que se encuentra la planta, para lo cual en la [Tabla 1.6](#) se muestran los efectos de iluminación en las plantas.

Tabla 1.6 Efectos de la Iluminación en las plantas [13]

Falta de luz	Exceso de luz
Las hojas caen, toman un color amarillento y se tiene un crecimiento tardado.	Hojas débiles y arrugadas con tonalidades marrones.
Se genera tallos sin color, delgados y excesivamente largos, con una notable distancia entre los entrenudos; además tienen poca vitalidad.	Las plantas pierden vigor en los momentos más calurosos del día.
Dificultad para que las plantas florezcan y si lo hacen, estas tienen flores incompletas o son muy pequeñas cayendo antes de florecer correctamente.	Las hojas pierden color tomando tonalidades blanquecinas.

En la [Tabla 1.7](#) se encuentra rangos de iluminancia en diferentes condiciones y como estos valores afectan positiva o negativamente a las plantas.

Tabla 1.7 Influencia de distintos rangos de iluminancia en las plantas [14]

Iluminancia (lux)	Influencia en las plantas
0-200	Es inadecuada para todas las plantas.
200-500	Es inadecuada para casi todas las plantas.
500-1000	Intensidad de luz adecuada solo para plantas que crecen en sombra.
1000-2000	Intensidad de luz adecuada para plantas que requieren luz solar indirecta.
2000-4000	Representa a un 40% de la iluminancia en el sol al medio día y es apropiada para plantas que requieren de intensidad de luz constante y de calidad.
4000-5000	Representa a un 50% de la iluminancia en el sol al medio día y es apropiada para plantas que requieren luz fuerte.
5000-10000	Iluminancia muy brillante a la que aún se puede llegar en ambientes <i>indoor</i> , buena para plantas que requieren de altos niveles de luz.

1.4.2 Terapia Hortícola

La terapia hortícola es la habilidad de aprender y cuidar a las plantas con el fin de ayudar a personas con problemas de salud mental, adultos mayores, niños y personas que sufran de cuadros de depresión y ansiedad, aprovechando el contacto con la naturaleza [15].

Desde que la humanidad descubrió las propiedades medicinales que otorgan ciertas plantas, se ha relacionado al bienestar personal y al contacto con la naturaleza. El primer registro encontrado de personas realizando actividades hortícolas para uso terapéutico viene del antiguo Egipto, donde los médicos aconsejaban paseos por los jardines a los miembros de la realeza que padecían de discapacidades mentales [16].

En este contexto, mediante el prototipo implementado en este trabajo, se busca brindar asistencia a las personas que gusten de la jardinería y actividades relacionadas con el cuidado de las plantas, mejorando la forma en que se interactúa con las mismas, en la [Figura 1.3](#) se presentan jardines para meditación, relajación y necesidades terapéuticas, en los cuales se podría implementar el prototipo.



Figura 1.3 Jardines para meditación, relajación y necesidades terapéuticas [2].

1.4.3 Internet de las Cosas

El IoT es la nueva generación de innovación tecnológica, revolucionando la actividad económica de un país, donde empresas muy importantes están involucradas [17].

El Internet de las Cosas permite automatizar varios aspectos de la vida cotidiana, ayudando a sectores económicos como la agricultura, industria y medicina [18].

1.4.3.1 Automatización a la Agricultura mediante Sistemas IoT

La automatización está estrechamente relacionada al Internet de las Cosas, generando soluciones rentables y eficientes a diversas necesidades de la población [4]. Se espera que en el 2023 en América Latina, el sector del Internet de las Cosas tenga cerca de 996 millones de dispositivos conectados [19].

La automatización permite reducir tiempo, recursos y errores humanos en la agricultura, dado que se permite controlar el cultivo y la calidad del suelo de manera remota con cualquier dispositivo que tenga acceso a Internet [4]. Se puede realizar un análisis del estado de las plantas, comunicando al encargado por medio de una plataforma o aplicación, como se encuentra el suelo, en donde las plantas fueron cultivadas o si los niveles de iluminación y humedad son los adecuados para esa plantación.

Al observar las necesidades que pueden existir para invernaderos, jardines y los diferentes tipos de agricultura existentes, se comprende cómo las condiciones ambientales en las que se encuentra una planta afectan directamente a su crecimiento. Dependiendo de las necesidades específicas de cada caso se procede a valorar que parte del proceso de crecimiento de las plantas debe ser automatizado [4].

1.4.3.2 Sistemas Embebidos

Un sistema embebido es un sistema electrónico que realiza una o varias tareas específicas en tiempo real, generalmente como parte de un sistema más grande [20]. Los componentes del sistema embebido comúnmente están en la placa base y no tienen el aspecto que generalmente se asocia a una computadora [1].

Los sistemas embebidos se programan dependiendo de la placa utilizada, haciendo uso de lenguaje C, C++ o JAVA [21]. Algunos ejemplos de sistemas embebidos son: Raspberry Pi, BeagleBone y Arduino.

1.4.3.3 Sensores utilizados en la Agricultura

Los sensores son dispositivos que responden a un cambio que ocurre al alterar una variable, transformando los valores obtenidos a una señal eléctrica la cual es más fácil de interpretar, siendo este un valor de medida [22].

Al recibir un estímulo los sensores reaccionan generando una señal eléctrica, donde las señales que se obtienen a la salida son señales de voltaje o corriente. Los sensores pueden acoger varios tipos de señales como señales químicas, bioquímicas o físicas, transformando éstas a señales eléctricas [23].

Los sensores se pueden clasificar según su aplicación, su mecanismos de conversión, el tipo de señal de entrada (Analógica/ Digital), el material del que están hechos y las características que posee el sensor, valorando parámetros como su precisión, el rango en el que actúa y su costo [23]. En la [Tabla 1.8](#) se presentan diferentes tipos de sensores con su descripción.

Tabla 1.8 Tipos de sensores y su descripción [24].

Tipo de sensor	Descripción
Sensores pasivos	La señal de salida es generada casi completamente por la señal media sin tensión de excitación.
Sensores activos	Estos sensores dependen de una fuente de alimentación externa, generando la mayor parte de la señal de salida del sensor.
Sensores analógicos	El sensor produce una señal continua y proporcional a la cantidad medida.
Sensores digitales	El sensor produce una señal en forma binaria

1.4.3.4 Sensores de Humedad en el Suelo

Para la detección de humedad en el suelo se tiene varias alternativas como se observa en la [Tabla 1.9](#). Incluir también un sensor de temperatura incrementa la precisión, dado que la resistencia del suelo también depende de la temperatura. Otra alternativa es hacer uso de un sensor de humedad dieléctrico, estos sensores son básicamente condensadores en los que la tierra responde como la parte dieléctrica del condensador por lo que la constante dieléctrica varía al incrementar la humedad [25].

Tabla 1.9 Tipos de sensores para medir humedad del suelo [25]

Tipo de sensor	Características
Tensiómetro	<ul style="list-style-type: none"> • Se tiene incorporado un vacuómetro dentro de un tubo de plástico y mide la presión. • Mide los cambios de presión al sacar agua del suelo. • Al tener una lectura alta se tiene una humedad baja y viceversa.
Fibra óptica	<ul style="list-style-type: none"> • Mide la variación de intensidad de luz que viaja de una sonda al suelo y la intensidad de luz que regresa del suelo a la sonda.
Bloques de Resistencia Eléctrica (Bloques de Yeso) y Matriz Granular (Filigrana)	<ul style="list-style-type: none"> • Se usa dos electrodos en un material poroso, como lo es el yeso y se mide la resistencia entre sí. • El agua que se desplaza hasta el yeso disminuye la resistividad del suelo y cuando se extrae agua la resistividad aumenta. • Tener una resistencia baja significa humedad alta y una resistencia alta significa humedad baja.
Sondas de conductividad eléctrica	<ul style="list-style-type: none"> • Se mide la corriente entre dos sondas, con contacto directo con el suelo. • Tener una conductividad alta significa una humedad alta y viceversa.
Disipación de calor	<ul style="list-style-type: none"> • Se tiene un medio cerámico en el sensor que mide el calor que disipa el suelo. • A mayor disipación se tiene mayor nivel un nivel de humedad alta y viceversa.
Dieléctrico: Capacitancia	<ul style="list-style-type: none"> • Se tiene dos electrodos dieléctricos en contacto con el suelo, aplicándose una frecuencia de oscilación alta a los electrodos para así medir la frecuencia de resonancia. • Según el nivel de humedad del suelo varía la frecuencia de resonancia. • Al tener un cambio de frecuencia alto se tendrá un nivel de humedad alto y viceversa.
Dieléctrico: TDR (Reflectometría en el dominio del tiempo)	<ul style="list-style-type: none"> • Se realiza una medición del viaje en el tiempo de un pulso electromagnético a lo largo de la varilla de la sonda. • A mayor tiempo de viaje se tendrá un nivel de humedad alto.

1.4.3.5 Sensores de Luz ambiental

Para detectar el nivel de luz ambiental se tiene diferentes tipos de sensores de luz, siendo los sensores presentados en la [Tabla 1.10](#) los más económicos y simples disponibles en el mercado, permitiendo así que estos sensores sean muy utilizados en productos de consumo, agricultura, cámaras y luces nocturnas [26].

Tabla 1.10 Funcionamiento de los diferentes tipos de sensores de luz ambiental

Tipo de sensor	Modo de funcionamiento
Fotorresistencia	Resistencia eléctrica que varía el valor dependiendo de la cantidad de luz a la que este expuesta la superficie, a mayor se la intensidad de la luz menor será el valor de la resistencia y viceversa [27].
Fotodiodo	Semiconductor P-N que reacciona a la presencia de luz infrarroja y visible. El fotodiodo es una combinación de una fotorresistencia y un diodo [28].
Fototransistor	Transistor en el que la unión Base-Colector es sensible a la luz, donde se convierte los fotones recibidos en corriente de baja potencia y permite la conducción de corriente entre Colector y Emisor generando corriente [29].
Fotodiodo y amplificador de corriente	La corriente producida por el fotodiodo es amplifica en un amplificador de transimpedancia transformándose en un voltaje de salida [30].

1.4.4 Plataformas para Monitoreo de Datos utilizando Sistemas Embebidos

Para observar la información generada por los sensores se hace uso de plataformas para proyectos IoT que gestionan los datos en la nube. En la [Tabla 1.11](#) se muestra varias plataformas usadas en proyectos IoT [31].

Tabla 1.11 Comparación de plataformas para el desarrollo de proyectos IoT [31][32]

	Microcontroladores	Forma de conexión	Ventaja
Blynk	ESP32, BBC Micro:bit, ESP8266, ESP8266, Lantronix Fox 3, Microduino, Onion Omega, Particle, Pycom WiPy, Raspberry Pi.	Wi-Fi, Ethernet y Puerto serial.	Genera aplicaciones móviles sin código para IoT.
aREST Framework	Raspberry Pi, Arduino, ESP8266.	Wi-Fi, Puerto serial, Bluetooth y Ethernet.	Brinda soporte para realizar aplicaciones con servicio RESTful.
Thingier.io	ESP8266, Raspberry Pi, Intel Edison.	WiFi, Bluetooth, Puerto serial.	Programación sencilla y ofrece cuentas gratuitas.
Arduino Cloud	Arduino, ESP 32, ESP8266.	WiFi, Puerto serial.	Fácil manejo de la plataforma.
ThingSpeak	Arduino, ESP 32, ESP8266.	WiFi, Bluetooth.	Enfocada a aplicaciones IoT.

2 METODOLOGÍA

En este capítulo se detalla la forma de implementación del prototipo para la detección del nivel de humedad del suelo e iluminación de un jardín, identificando la metodología con la que se trabajó, los elementos utilizados, su forma de conexión, la programación y todas las actividades para cumplir con los objetivos planteados.

2.1 Descripción de la Metodología utilizada

Para la realización del prototipo se optó por utilizar la metodología cascada, donde se permite el desarrollo tareas secuenciales para la realización de proyectos basados en hardware y software. Esta metodología trabaja por etapas, promoviendo la filosofía paso a paso realizada por bloques de tareas. Su nombre está dado por las fases que la conforman, pues se colocan de arriba hacia abajo siguiendo un orden secuencial como se muestra en la [Figura 2.1](#) [33].



Figura 2.1 Metodología cascada [33]

- **Fase 1: Requisitos**

Durante esta fase, se lleva a cabo un análisis de los requisitos necesarios para la implementación del prototipo. Se evalúan los componentes de hardware y software que se utilizarán, teniendo en cuenta posibles limitaciones, servicios brindados y metas existentes.

- **Fase 2: Diseño**

Durante la fase de diseño del prototipo, se define la arquitectura del sistema, la cual muestra la relación entre los componentes de hardware y software. En esta etapa, se establece la forma en que el usuario interactuará con los diferentes componentes del prototipo.

- **Fase 3: Implementación**

En esta fase se implementa el prototipo y se ejecutan todas las tareas que debería cumplir el mismo, verificando que tanto la parte de software como de hardware funcionen adecuadamente y se encuentre todo en excelentes condiciones.

- **Fase 4: Verificación**

Antes de obtener el producto final, es fundamental verificar el correcto funcionamiento de todas las partes que conforman el prototipo. Para ello, es necesario corroborar los valores obtenidos tanto para la detección de humedad en el suelo como para la iluminación en el ambiente. Se genera diferentes escenarios de prueba y se da los últimos ajustes al prototipo de ser necesario.

Fase 5: Mantenimiento

La fase de mantenimiento es crucial en el ciclo de desarrollo, ya que se realiza después de poner en marcha el prototipo en un jardín residencial o una planta específica. Durante esta fase, se detectan y corrigen errores que no fueron descubiertos en etapas anteriores. Además, se adapta y mejora el prototipo para cumplir con los nuevos requerimientos que surgen con el tiempo. Es importante destacar que esta fase es la más larga del ciclo de desarrollo y requiere de una atención constante para garantizar el óptimo funcionamiento del prototipo [33].

2.2 Requerimientos de Hardware para el Prototipo

En esta sección se describe cada elemento de hardware utilizados para generar el prototipo. Además, se indica en qué contribuye cada uno de estos elementos en el prototipo desarrollado.

Se utiliza el módulo ESP32 por su conectividad inalámbrica Wi-Fi, tamaño reducido y bajo costo; además, de que es compatible con el IDE de Arduino. Se complementa el prototipo con el uso del sensor capacitivo de humedad en el suelo v1.2 para la detección de humedad en el suelo y el sensor de luz BH1750 para la detección de iluminación en el ambiente.

2.2.1 Módulo Wi-Fi ESP32

El ESP32 es diseñado para dispositivos electrónicos portátiles, aplicaciones móviles e Internet de las Cosas (IoT), donde tiene todas las características principales de chips de baja potencia, como la activación de reloj de granularidad fina, escalado dinámico de alimentación y múltiples modos de alimentación. Posee una salida ajustable del

amplificador de potencia lo que promueve un compromiso óptimo entre el rango de transferencia de datos, la velocidad de envío de los datos y el consumo de energía [34].

La [Figura 2.2](#) muestra este chip donde su característica más notable es que incluye Wi-Fi y permite el uso de Bluetooth. Se basa en la tecnología de bajo consumo de 40 nm de TSMC. La entrada analógica del ESP32 transforma los voltajes entre 0V y 3.3V a valores entre 0 y 4095. Tiene una amplia gama de posibles aplicaciones [35][34].

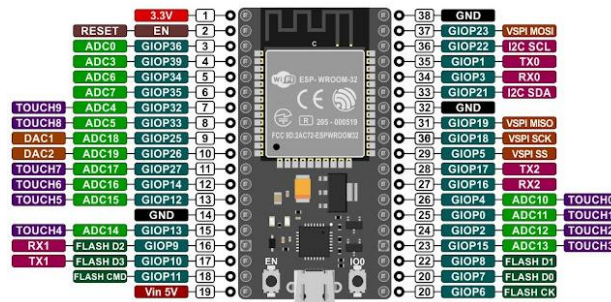


Figura 2.2 Módulo Wi-Fi ESP32 y su distribución de pines [36]

En la [Tabla 2.1](#) se comparan las placas Arduino Uno, ESP8266 y ESP32 para entender por qué trabajar con el ESP32 es la mejor opción.

Tabla 2.1 Comparación entre las características de Arduino Uno, ESP8266 y ESP32 [37]

	Arduino Uno	ESP8266	ESP32
Numero de núcleos	1	1	2
Arquitectura	8 Bit	32 Bit	32 Bit
Frecuencia CPU	16 MHz	80 MHz	160 MHz
Wi-Fi	No	Si	Si
Bluetooth	No	No	Si
RAM	2 KB	160 KB	512 KB
Flash	32 KB	16 MB	16 MB
Número de pines GIPO	14	17	36
Busses	SPI, 12C	SPI, 12C, UART, I2S	SPI, 12C, UART, I2S, CAN
Representación de los voltajes entre 0V y 3.3 en las entradas analógicas	0-1028	0-1028	0-4095
Pines ADC	6	1	18
Pines DAC	0	0	2

Los paquetes de software disponibles para el ESP32 permiten que pueda ser usado con Arduino IDE como entorno de desarrollo; además, es importante considerar que las bibliotecas utilizadas en el ESP8266 y en Arduino siguen utilizándose en la ESP32, no obstante, se recomienda consultar el foro ESP32 en donde se puede aprender a programar el ESP32 en el IDE de Arduino [27].

2.2.2 Sensor Capacitivo de Humedad en el Suelo v1.2

El sensor de humedad en el suelo ([Figura 2.3](#)) permite medir la intensidad de humedad en el suelo por detección capacitiva. Está fabricado de materiales que son anticorrosivos por lo que tienen larga vida útil [38]. Es un sensor analógico que posee un regulador de voltaje para trabajar entre 3.3V y 5.5V DC con una corriente de operación de 5mA, voltaje de salida DC entre 0V y 3V . Es muy utilizado en la agricultura inteligente y jardinería inteligente [38].



Figura 2.3 Sensor capacitivo de humedad en el suelo v1.2 [39]

2.2.3 Sensor de Luz digital BH1750

El sensor BH1750 ([Figura 2.4](#)) mide la iluminancia en una superficie en lux (*lumenes/m²*) con un rango de medición entre 1 lux y 65535 lux. Este utiliza el protocolo I2C (*Inter-Integrated Circuit*) por lo que es sencillo usarlo con microcontroladores como el ESP32. Se emplean los pines SCL (*Serial Clock*) y SDA (*Serial Data*) en su conexión y no se necesita realizar cálculos adicionales para obtener los valores en lux pues el sensor BH1750 proporciona directamente los valores en lux lo cual es una gran ventaja a comparación con otros sensores de luz como por ejemplo LDR [40].

Para alimentar el sensor se debe tener un voltaje entre 2.4V y 3.6V con una corriente de 0.12mA. Otra característica importante del sensor BH1750 es que los resultados otorgados no están sujetos a la fuente de luz utilizada como luz de día, led, incandescente o luz

halógena. Además, la radiación IR no es tan relevante para los resultados y el promedio de error es muy bajo estando entre un +/-20% [40].



Figura 2.4 Sensor de Luz digital BH1750 [41]

2.2.4 Relé HK19F DC5V-SHG

Es un dispositivo electromagnético que funciona como un interruptor, cerrando o abriendo el paso de corriente al accionar sus contactos eléctricamente. El Relé HK19F ([Figura 2.5](#)) tiene ocho pines y dos contactos utilizando una sola bobina para su control. Se tiene tres pines por cada contacto, el NC (Normalmente cerrado), COM (Común) y NO (Normalmente abierto) y en la [Tabla 2.2](#) se encuentra las características de funcionamiento del Relé HK19F [42].



Figura 2.5 Relé HK19F DC5V-SHG [43]

Tabla 2.2 Características de funcionamiento del Relé HK19F DC5V-SHG [42]

Característica	Valor
Voltaje de entrada a la bobina	5V-DC
Corriente de contacto a 30V-DC	2A
Corriente de contacto a 125V-AC	1A

2.2.5 Mini bomba de agua sumergible 3-6V

La bomba de agua que se muestra en la [Figura 2.6](#) es capaz de mover dos litros de agua por minuto, con un voltaje de entrada entre 3V a 6V, es completamente sumergible y comúnmente utilizada para el riego de plantas o fuentes decorativas [44].

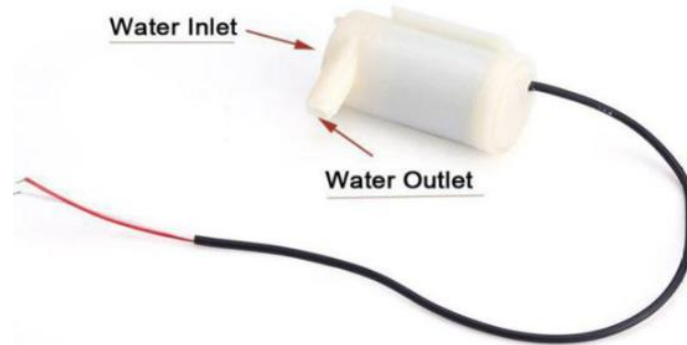


Figura 2.6 Mini bomba de agua sumergible 3-6V [45]

2.3 Requerimientos de Software para el Prototipo

En esta sección se describe el componente de software utilizados para implementar el prototipo. Además se indica en que contribuye cada uno de estos elementos en el prototipo desarrollado. Se hace uso de la plataforma Arduino IoT Cloud, para que a través de una interfaz gráfica se muestren los valores obtenidos por los sensores gráficamente.

2.3.1 Arduino IoT Cloud

La plataforma Arduino IoT Cloud permite realizar proyectos relacionados con IoT, proporciona una interfaz gráfica autogenerada por la plataforma según las necesidades del usuario y los componentes que se utilicen. Arduino IoT Cloud permite manejar dispositivos IoT, así como configurar, crear, cargar, visualizar o codificar proyectos IoT y sus características se presentan en la [Tabla 2.3](#) [46].

2.3.2 IFTTT

IFTTT es una plataforma generadora de rutinas para relacionar distintos servicios o productos inteligentes, para lo cual se usa el principio "Si esto, entonces aquello" o en inglés "*If This, Then That*" que es de donde viene las siglas de la plataforma. Permite programar acciones con distintas herramientas y aplicaciones automáticamente, donde si se realiza una acción en una de ellas otra acción será ejecutada automáticamente en otra aplicación o en la misma aplicación [47].

Tabla 2.3 Características de Arduino IoT Cloud [46]

Característica	Descripción
Sincronización de variables	Permite la sincronización de variables entre dispositivos, generando comunicación entre ellos, sin la necesidad de generar códigos extensos.
Monitoreo de datos	Monitoreo de los valores otorgados por los sensores que han sido conectados al microcontrolador.
Cargas por aire (OTA)	Facilidad para cargar el código a dispositivos no conectados directamente por el puerto serial de la computadora.
Programador	Permite activar o desactivar tareas en periodos de tiempo determinados.
Webhooks	Integración de proyectos IoT con diferentes servicios, como los es IFTTT (<i>If This, Then That</i>).
Uso compartido del panel	Se permite el uso compartido de los datos generados en la aplicación con otros usuarios de todo el mundo.
Soporte de Amazon con Alexa	Se tiene la integración con el asistente Alexa para cualquier proyecto que sea implementado en esta plataforma.
Hardware compatible	Las placas compatibles con Arduino IoT Cloud son las oficiales de Arduino o las placas ESP866 y ESP32 .
Conexiones admitidas	Se permite la conexión por LoRaWAN (con la ayuda de The Things Network), redes móviles y Wi-Fi.

Por ejemplo, con IFTTT se puede automatizar rutinas de la vida cotidiana como encender la cafetera automáticamente al detectar que se han encendido las lucen en una habitación, que al ir a descansar se apaguen las luces automáticamente o realizar envío de alertas [47] [48].

Se tienen servicios compatibles con IFTTT combinables entre sí, donde se encuentran también las redes sociales y plataformas más populares en el mercado como Facebook, Instagram, Twitter, Google o Spotify. En la [Figura 2.7](#) se muestra un ejemplo de como IFTTT trabaja con otros sistemas [47].

2.3.3 EasyEDA

EasyEDA es una herramienta en línea gratuita para el desarrollo de sistemas electrónicos que incluye simulación de circuitos y generación de placas de circuito impreso (PCB). Ofrece la facilidad de uso sin necesidad de descargar o instalar el software en el ordenador,

y cuenta con una amplia gama de herramientas para el diseño de circuitos esquemáticos y la simulación de circuitos digitales, analógicos y con señal mixta [50].

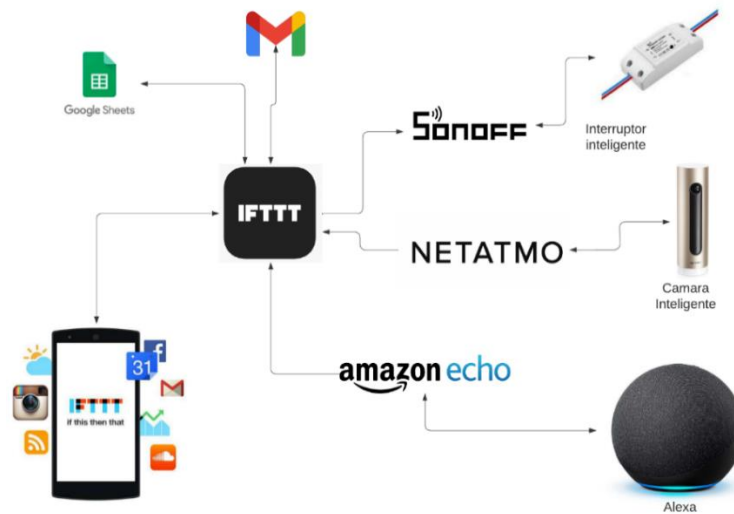


Figura 2.7 Ejemplo para la arquitectura de IFTTT [49]

La principal ventaja de EasyEDA es su extensa biblioteca dado que se tiene a disposición miles de componentes para el desarrollo de sistemas electrónicos y siempre se está actualizando con los nuevos componentes electrónicos que ingresan al mercado, todo esto para diseño de circuitos y modelación de circuitos impresos [50].

2.4 Diseño del Prototipo

La [Figura 2.8](#) presenta el diagrama planteado para el prototipo de detección de humedad de suelo e iluminación de un jardín. Describe como el sensor capacitivo de humedad en el suelo v1.2 y el sensor de luz BH1750, toman los niveles de humedad en el suelo e iluminación respectivamente en el ambiente de un jardín o planta. Posteriormente con los valores obtenidos a través del ESP32m, se transferieren a la plataforma Arduino IoT Cloud donde serán visualizados y monitoreados por el usuario.

Una parte relevante del prototipo es que además de monitorear los datos, cuando pasan determinados umbrales de humedad e iluminación se activa una alerta y un actuador dependiendo de las necesidades del jardín. Si se detecta un nivel de humedad en el suelo inadecuado y las plantas no puedan absorber los nutrientes necesarios, se procede a activar una bomba de agua, que dará riego a las plantas que lo necesiten. Por otra parte, si el nivel de iluminación en el jardín es bajo o se encuentra completamente en la oscuridad se envía una alerta al usuario vía email con la ayuda del protocolo HTTPS y la página web IFTTT, la cual permite programar el contenido del correo que será enviado.

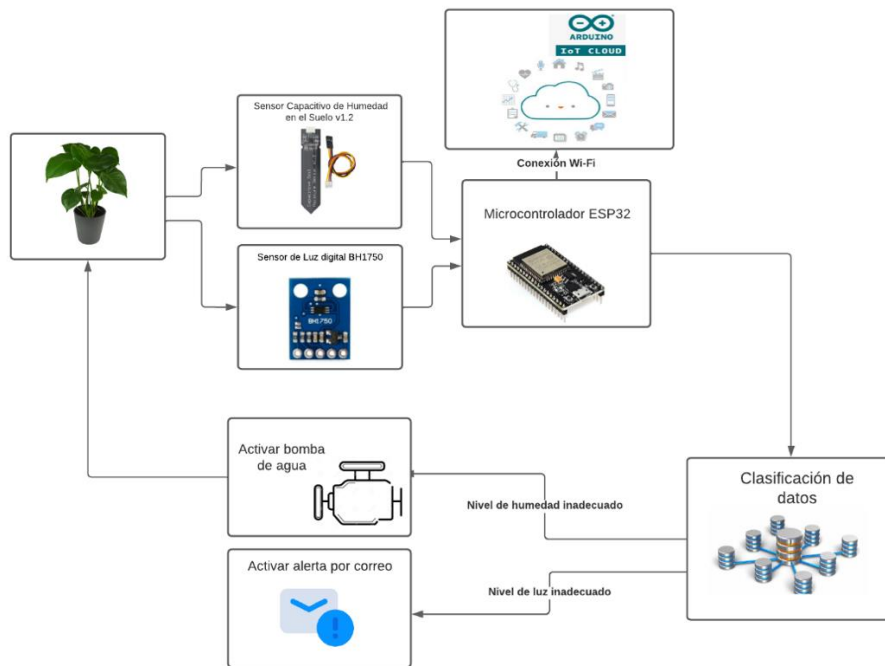


Figura 2.8 Diagrama de funcionamiento del prototipo

2.4.1 Diseño del Circuito Esquemático

Para el diseño del circuito esquemático mostrado en la [Figura 2.9](#) se utiliza la herramienta de software EasyEDA la que permite dar un primer vistazo de cómo queda el circuito diseñado y la forma de conexión entre los pines de cada elemento que esta detallada en la [Tabla 2.4](#).

2.4.2 Comunicación del ESP32 con la plataforma de Arduino Cloud IoT

Este proceso se lo realiza mediante Wi-Fi y se presenta un tutorial paso a paso de este proceso en el [Anexo III](#), generando el [Código 2.1](#) el cual permite conectar el ESP32 a la plataforma de Arduino IoT Cloud.

Tabla 2.4 Pines analógicos y digitales del ESP32 utilizados para el prototipo

PIN del ESP32	Función
GIOP34	Entrada de los valores analógicos proporcionados por el sensor capacitivo de humedad en el suelo v1.2.
GIOP22 (SCL)	Pin de reloj en serie (SCL) que el ESP32 pulsa a intervalos regulares.
GIOP21 (SDA)	Se envía los datos entre los dispositivos y se obtiene los valores de iluminancia proporcionados por el sensor BH1750.
GIOP18	Salida analógica para controlar la bomba de agua.

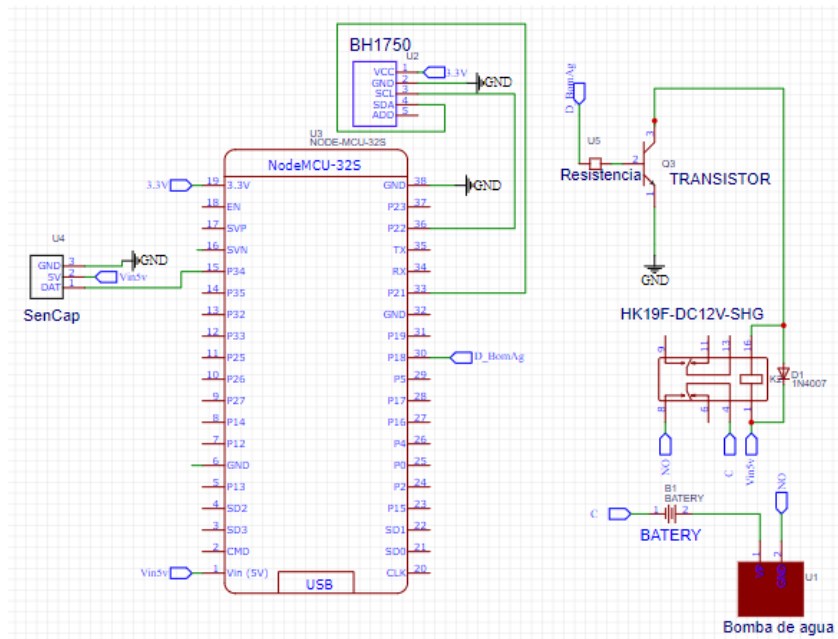


Figura 2.9 Circuito Esquemático del prototipo

Código 2.1 Conexión del ESP32 con Arduino IoT Cloud

```
//Librerias
#include <ArduinoIoTCloud.h>
#include <Arduino_ConnectionHandler.h>

WiFiMulti wifiMulti;

const char DEVICE_LOGIN_NAME[] = "8e726d43-5345-49ed-b6e4-fb7f11eedfe5";
// Nombre del inicio de sesión del dispositivo (generado automáticamente)

const char SSID[] = "Nombre_de_Red"; // Red SSID (nombre)
const char PASS[] = "Clave_de_Red"; // Contraseña de red
// usar para WPA o usar como clave para WEP
const char DEVICE_KEY[] = "PL06CESKFOAASKVXNXJQ"; // Contraseña secreta
del dispositivo ESP32
int humedad;
int luminosidad;

void initProperties(){

  ArduinoCloud.setBoardId(DEVICE_LOGIN_NAME);
  ArduinoCloud.setSecretDeviceKey(DEVICE_KEY);
  ArduinoCloud.addProperty(humedad, READ, ON_CHANGE, NULL);
  ArduinoCloud.addProperty(luminosidad, READ, ON_CHANGE, NULL);
}
WiFiConnectionHandler ArduinoIoTPreferredConnection(SSID, PASS);
```

2.4.3 Comunicación del Hidrómetro con el ESP32

El sensor capacitivo de humedad en el suelo v1.2 envía una señal analógica al ESP32 por el pin analógico GIOP34 como se muestra en la [Figura 2.10](#) y en la [Tabla 2.5](#). Al realizar las correspondientes pruebas se tiene que al recibir un valor entre 2000 y 2800 indica un suelo húmedo, y entre 3020 y 4095 un suelo seco, por lo que al tener un valor de entrada analógico al ESP32 de 2000 se tiene un 100% de humedad en el suelo y al tener un valor de 4095 se tiene un 0% de humedad en el suelo. Esto se encuentra expresado en el [Código 2.2](#), donde se ha mapeado los valores para obtener el porcentaje adecuado de humedad en el suelo.

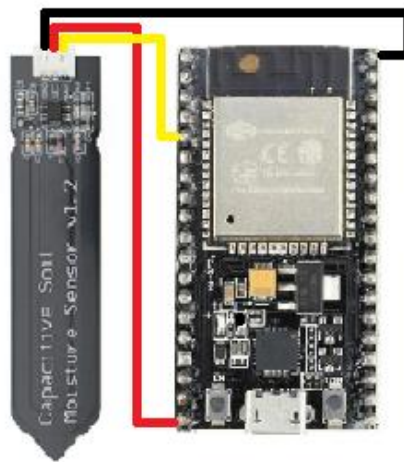


Figura 2.10 Modo de conexión del sensor de humedad en el suelo v1.2 con el ESP32

En la [Tabla 2.5](#) se encuentra especificada la forma de conectar el sensor capacitivo de humedad en el suelo v1.2 con el ESP32.

Tabla 2.5 Conexión de los pines del ESP32 con los pies del sensor capacitivo de humedad en el suelo v1.2

PIN del ESP32	PIN del Sensor de Humedad
GIOP34	AOUT
Vin 5V	VCC
GND	GND

Código 2.2 Sección del código para la obtención del % de humedad en las plantas

```
int humedad;
int SenCap = 34;
int HUM;
int ValHum;
const int ValAire = 4095;
//Valor obtenido por el sensor de humedad en el suelo en el aire
const int ValAgua = 2000;
//Valor obtenido por el sensor de humedad en el suelo en el agua
void loop() {
//Valores finales de humedad
HUM = analogRead(SenCap);
ValHum = map(HUM, ValAgua, ValAire, 100, 0);

//IMPRESIÓN DE DATOS
if(ValHum >= 100)
{
    humedad = 100;
    Serial.print("Humedad del suelo:
");Serial.print(humedad);Serial.println(" %");
}
else if(ValHum <=0)
{
    humedad = 0;
    Serial.print("Humedad del
suelo:");Serial.print(humedad);Serial.println(" %");
}
else if(ValHum >0 && ValHum < 100)
{
    humedad = ValHum;
    Serial.print("Humedad del suelo:
");Serial.print(humedad);Serial.println(" %");
}
delay(2000);//pausa de 2 segundo solo para ver de mejor forma los valores
expuestos en el monitor serial
}
```

2.4.4 Comunicación del Sensor de Luz digital BH1750 con el ESP32

El sensor de luz BH1750 envía la señal digital a los pines I2C (SCL y SDA) del ESP32 como se ve en la [Figura 2.11](#) y en la [Tabla 2.6](#). Para la obtención de datos con este sensor se utiliza la librería BH1750 de *Christopher Laws*. No se requiere de ninguna ecuación adicional para obtener los valores deseados pues da los valores directamente en luxes. En el [Código 2.3](#) se presenta la forma de obtener la iluminación en luxes.

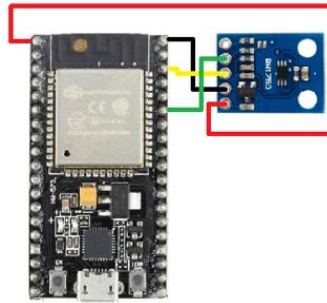


Figura 2.11 Modo de conexión del sensor de luz BH1750 con el ESP32

Tabla 2.6 Conexión de los pines del ESP32 con los pines del sensor de luz BH1750

PIN del ESP32	PIN del Sensor de Luz BH1750
GIOP22	SCL
GIOP21	SDA (datos)
3.3 V	VCC
GND	GND

Código 2.3 Sección del código necesaria para la obtención la iluminancia en luxes

```
#include <Wire.h>
#include <BH1750.h>

BH1750 lightMeter;

void setup() { // Se inicializa el bus I2C
  Wire.begin();
  lightMeter.begin();
  Serial.println(F("Prueba bh1750 inicializada"));
}

void loop() {
  //Valores finales de Iluminancia
  luminosidad = lightMeter.readLightLevel();
  //IMPRESIÓN DE DATOS
  Serial.print("Iluminancia: ");Serial.print(luminosidad);Serial.println("
lux");
  delay(2000); //pausa para ver los valores expuestos en el monitor serial
}
```

2.4.5 Activación de Bomba de Agua para Suelos Secos

Para el desarrollo de este prototipo y en específico para la activación automática de la bomba de agua, se opta por utilizar un relé simple controlado por un transistor común. Se analizó la forma de conexión de cada componente y como estos funcionan en el circuito

electrónico, el circuito utilizado para poder controlar el relé con las salidas digitales del ESP32 es el de la [Figura 2.12](#) [51].

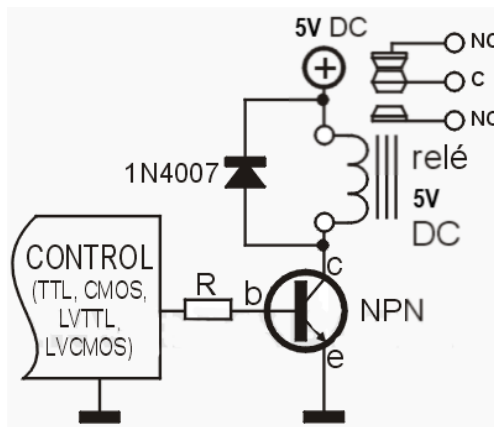


Figura 2.12 Relé controlado por transistor con niveles de tensión TTL, CMOS, LVTTTL y LVCMOS [51]

Para el control de un relé se utiliza un transistor NPN conectado como se ve en la [Figura 2.12](#). De esta manera el transistor cierra el circuito al conectar el voltaje de entrada DC en un terminal de la bobina y al colector del transistor el otro terminal de la bobina del relé. El usar este tipo de circuitos brinda la posibilidad de usar relés con voltajes de alimentación distintos, donde el circuito de control trabaja con componentes lógicos que utilizan tecnología TTL (5V), LVTTTL (3.3V), CMOS (5V y 15 V) y LVCMOS (3,3V, 2,5V o 1,8V) [51].

Para la selección de componentes se realiza los siguientes pasos [51].

1) Elección del Relé

EL relé utilizado por el prototipo es el Relé HK19F DC5V mostrado en la [Figura 2.5](#) el cual trabaja con un voltaje de 5V en la bobina y una corriente de hasta 2A.

2) Elección del tipo de transistor

Considerando los niveles del voltaje y corriente a los que trabaja el Relé HK19F y que los niveles mínimos de corriente para activar el relé oscilan entre los 20 mA y los 100 mA, donde para corrientes menores a los 100 mA se puede utilizar cualquier transistor NPN común. Por esto se escoge el transistor bipolar NPN 2N2222A ([Figura 2.13](#)) muy usado en conmutación de circuitos, teniendo una corriente en el colector buena [52].



Figura 2.13 Transistor bipolar NPN 2N2222A [53]

3) Protección del transistor a corrientes inversas

En el momento que se desactiva el relé existe riesgo de quemarlo, dado que se suspende la corriente a la bobina por lo que se genera un campo magnético en un breve instante de tiempo, lo que genera un voltaje alto de polaridad inversa en los terminales, corriendo el peligro de dañar el transistor.

Por esto se utiliza un diodo rectificador 1N4007 ([Figura 2.14](#)) el cual se encuentra polarizado inversamente y en paralelo con la bobina.



Figura 2.14 Diodo rectificador 1N4007 [54]

4) Calculo de la resistencia en base del transistor

Inicialmente se debe considerar que para que el transistor conduzca, se requiere de una tensión superior a 0,6V en la base del transistor y así la corriente pueda pasar entre emisor y colector. Por lo que se puede utilizar sin problemas niveles lógicos con valores de 1.8V, 2.5V, 3.3V, 5V y 12V.

Para el cálculo de la resistencia en base del transistor se tiene la [Ecuación 2.1](#).

$$R = \frac{(V_{in} - 0,6) * HFE}{I_{Relé}}$$

Ecuación 2.1

Donde:

R: Resistencia en base (Ω)

HFE: Ganancia en continua del transistor

$I_{Relé}$: Corriente propia del relé (mA)

V_{in} : Voltaje de control (V)

Se considera los siguientes valores:

$$HFE = 200$$

$$I_{Relé} = 50 \text{ mA}$$

$$V_{in} = 3.3 \text{ V}$$

Con lo que la resistencia en base para conectar entre el transistor y el pin digita G10P18 del ESP32 es:

$$R = \frac{(3.3 \text{ V} - 0.6) * 200}{50 \text{ mA}} = 10.8 \text{ K}\Omega$$

Considerando las resistencias comerciales disponibles en el mercado se utiliza una resistencia $R = 10 \text{ K}\Omega$ mostrada en la [Figura 2.15](#).



Figura 2.15 Resistencia de $10 \text{ K}\Omega$

- 5) Finalmente se conecta los pines del Relé HK19F, el común (C) con el negativo de la fuente de alimentación por la bomba de agua y el normalmente abierto (NO) a la al GND de la bomba de agua. En la [Figura 2.16](#) se presenta la distribución de pines del Relé HK19F como guía para conectar cada elemento.

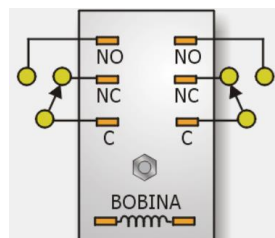


Figura 2.16 Distribución de pines del Relé HK19F visto desde la parte inferior [55]

La forma de conexión de la bomba de agua con todos los elementos utilizados se presenta la [Figura 2.17](#) y en el [Código 2.4](#) permite determinar el nivel de humedad en el suelo en el que se activa y se apaga la bomba de agua.

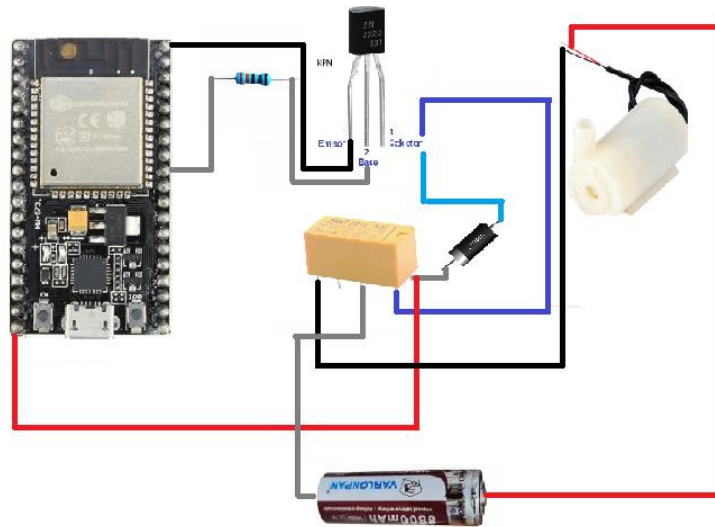


Figura 2.17 Modo de conexión de la bomba de agua con el ESP32

Código 2.4 Sección del código para la activación de la bomba de agua en suelos secos

```

int humedad;
int SenCap = 34;
int HUM;
int ValHum;

void loop() {
    //Valores finales de humedad
    HUM = analogRead(SenCap);
    ValHum = map(HUM, ValAgua, ValAire, 100, 0);

    if(ValHum >= 100)
    {
        humedad = 100;
    }
    else if(ValHum <=0)
    {
        humedad = 0;
    }
    else if(ValHum >0 && ValHum < 100)
    {
        humedad = ValHum;
    }
    //ACTUADOR PARA HUMEDAD EN EL SUELO
    if (humedad<21) // Nivel de encendido de la bomba de agua
    {
        Serial.println("Suelo seco - Bomba encendida");
        digitalWrite(218, HIGH); // señal en alto se enciende la
    }
    else if (humedad >= 40) // el nivel máximo de agua debe ser
    {
        Serial.println("Suelo mojado, Bomba apagada");
        digitalWrite(18, LOW); // señal en bajo se apaga la bomba de agua
    }
    delay(1000);
}

```

2.4.6 Activación de Alerta en Baja Iluminancia del Jardín

Como parte del del prototipo se establece que se enviará un correo automático al usuario encargado del cuidado del jardín, siempre que los niveles de iluminancia presentes en el jardín sean inferiores a un umbral establecido, alertando al usuario para que este realice las correcciones que correspondan. Para esto se utiliza la plataforma IFTTT, donde se genera una rutina para posteriormente ser activada por el módulo ESP32.

Para realizar el proceso de generación de rutinas a través de la plataforma IFTTT se presenta un tutorial en el [Anexo IV](#).

La solicitud web presentada es el evento generado y al ejecutar este link en un navegador se puede apreciar que llega un correo de alerta. Para activar la opción de envío mediante Arduino y el módulo ESP32 se presenta el [Código 2.5](#) el cual hace uso de HttpClient para realizar peticiones y recuperar respuestas.

Código 2.5 Sección del código para enviar un correo de alerta al detectar niveles bajos de iluminancia en el jardín

```
#include <Arduino.h>
#include <WiFi.h>
#include <WiFiMulti.h>
#include <HttpClient.h>
WiFiMulti wifiMulti;
int luminosidad;
BH1750 lightMeter;

String Url_LUZ =
"https://maker.ifttt.com/trigger/Sensor_iluminancia/json/with/key/j4ohQx8
yhUJU5jJeldbPMeF1R_gaU3M-XOvs1mM3yeW";
void setup() {

  for(uint8_t t = 4; t > 0; t--) {
    Serial.printf("CARGANDO %d...\n", t);
    Serial.flush();
    delay(1000);
  }
  wifiMulti.addAP(SSID, PASS);
}
void loop() {
  //Valores finales de Iluminancia
  luminosidad = lightMeter.readLightLevel();
  Serial.print("Iluminancia: ");Serial.print(luminosidad);Serial.println("
lux");
  delay(2000); //pausa de 2 segundo

  if (luminosidad<200) // Si la variable luminosidad es menor que 200 lux
se procede a realizar la peticion HTTP
  {
    if((wifiMulti.run() == WL_CONNECTED)) {
```

Código 2.5 Sección del código para enviar un correo de alerta al detectar niveles bajos de iluminancia en el jardín (continuación)

```
HTTPClient http;
Serial.print("Iniciando el envio de [HTTP] ...\\n");
http.begin(Url_LUZ); //Envia la peticion
Serial.print("Esperando respuesta de [HTTP] ...\\n");
int httpCode = http.GET();

if(httpCode > 0) {
  Serial.printf("Captura codigo [HTTP] ...: %d\\n", httpCode);
  if(httpCode == HTTP_CODE_OK) {
    String payload = http.getString();
    Serial.println(payload);
  }
} else {
  Serial.printf("Error: Captura fallida [HTTP] GET... %s\\n",
http.errorToString(httpCode).c_str());
}
http.end();
delay(30000); // El correo se envia cada 30 segundos mientras
se siga teniendo valores inadecuados de iluminancia
}
```

2.4.7 Alimentación del Circuito

Para alimentar el ESP32 se utiliza un cargador común de celular con 5V de salida ([Figura 2.18](#)), esto se puede realizar dado que dentro del ESP32 se tiene un regulador de voltaje, que transforma el voltaje entregado por el cargador a 3.3V que es el voltaje necesario para el correcto funcionamiento del ESP32.



Figura 2.18 Cargador que alimenta el ESP32

Por otro lado, para alimentar la bomba de agua se utiliza una pila recargable de 3.7 V ([Figura 2.19](#)), siendo este el voltaje adecuado para el funcionamiento de la bomba de agua

que necesita un voltaje de entrada constando de entre 3V a 6V como se indica en la [sección 2.2.5](#). Esta forma de alimentación se utiliza en lugar de sacar el voltaje del pin Vin del ESP32 directamente con el fin de precautelar daños en el módulo.



Figura 2.19 Pila recargable que alimenta la bomba de agua

2.4.8 Diagrama de Flujo

El código fuente que se realiza para la implementación del prototipo se presenta en el Anexo I, el cual se basa en el diagrama de flujo de la [Figura 2.20](#).

Para el diagrama inicialmente se establecen las variables a ser utilizadas y mediante las librerías implementadas el módulo ESP32 se conecta con la plataforma Arduino IoT Cloud, visualizando y monitoreando los valores del sensor capacitivo de humedad en el suelo v1.2 y el sensor de luz BH1750.

Si la adquisición de los datos generados por los sensores es exitosa, estos son introducidos a un operador *if*, que permiten establecer una alerta vía email si el valor de iluminancia es menor a 200 lux, se active como actuador la bomba de agua si el nivel de humedad en el suelo es menor al 21% y se apaga cuando el nivel de humedad sea mayor o igual al 40%.

La información transmitida tanto por el monitor serie como por la plataforma Arduino IoT Cloud seguirán siendo transmitidos hasta que se desconecte la fuente de alimentación del ESP32.

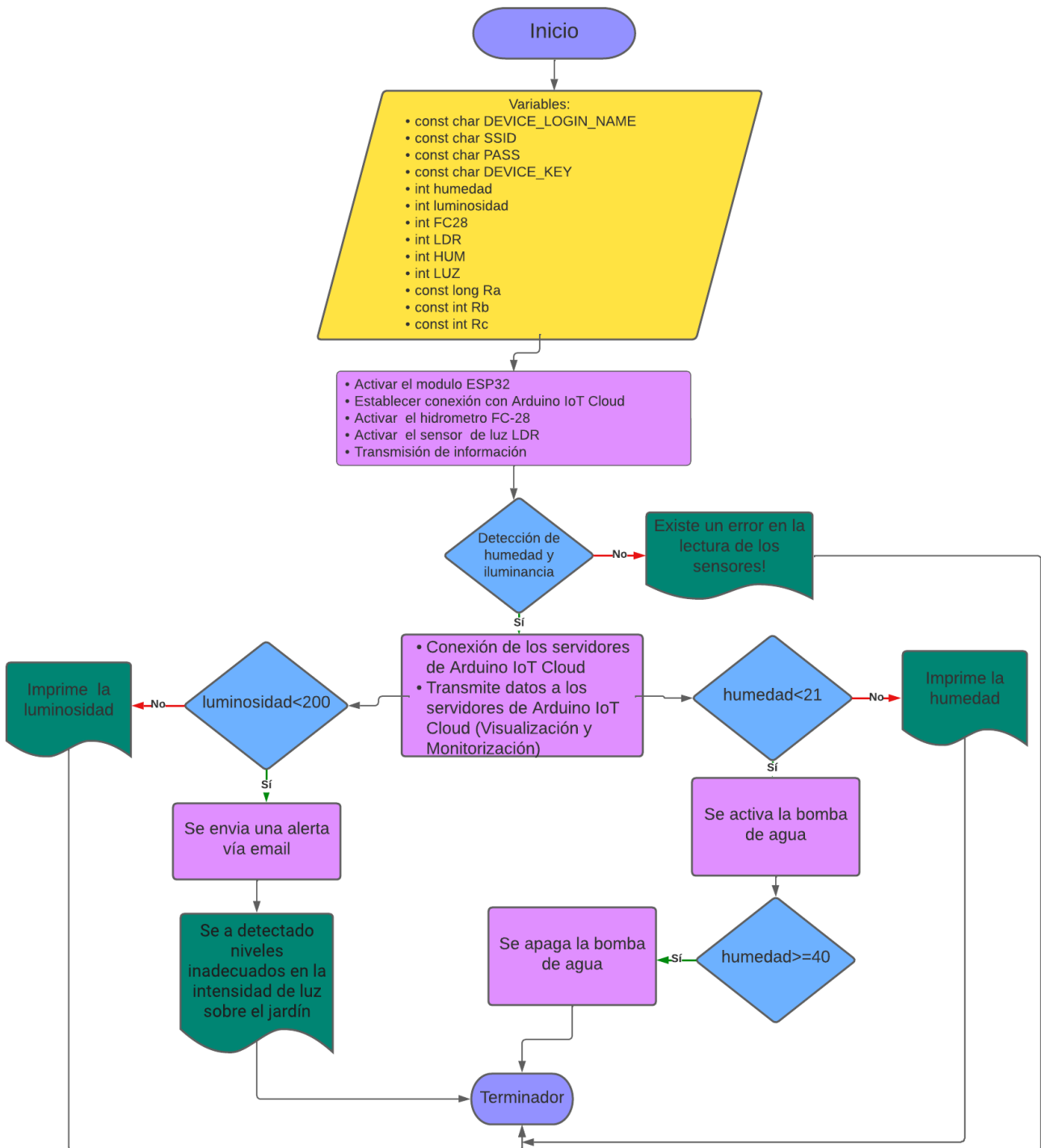


Figura 2.20 Diagrama de flujo para la detección del nivel de humedad del suelo e iluminación de un jardín

2.5 Implementación del prototipo

En esta sección se lleva a cabo la implementación de la placa de circuito impreso y el módulo protector al prototipo.

2.5.1 Desarrollo de la PCB

Para el diseño de la PCB se utiliza el software EasyEDA previamente detallado en la sección 2.3.3. Se utiliza una lámina de fibra de vidrio FR4 de dimensiones 13.2 x 8 (cm). Con las librerías proporcionadas por EasyEDA se recrea las dimensiones exactas para el sensor capacitivo de humedad en el suelo v1.2, el sensor de luz BH1750, el módulo ESP32 modelo NodeMCU-32S y la de todos los componentes electrónicos utilizados para la realización del prototipo. En la [Figura 2.21](#) se muestra la distribución de los componentes electrónicos utilizados para el prototipo. Además, en la [Figura 2.22](#) se presenta una visualización en 3D de como quedara la PCB en su capa inferior y superior.

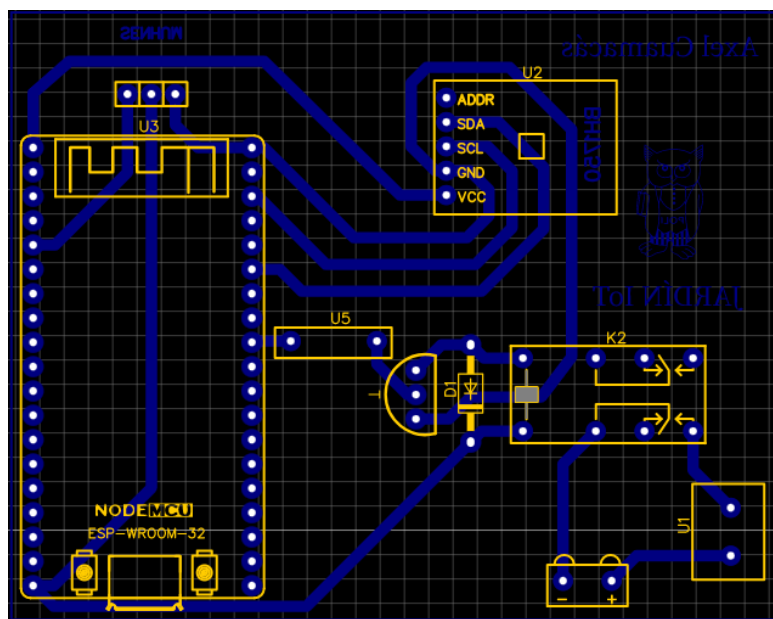


Figura 2.21 Distribución de los componentes electrónicos en la PCB

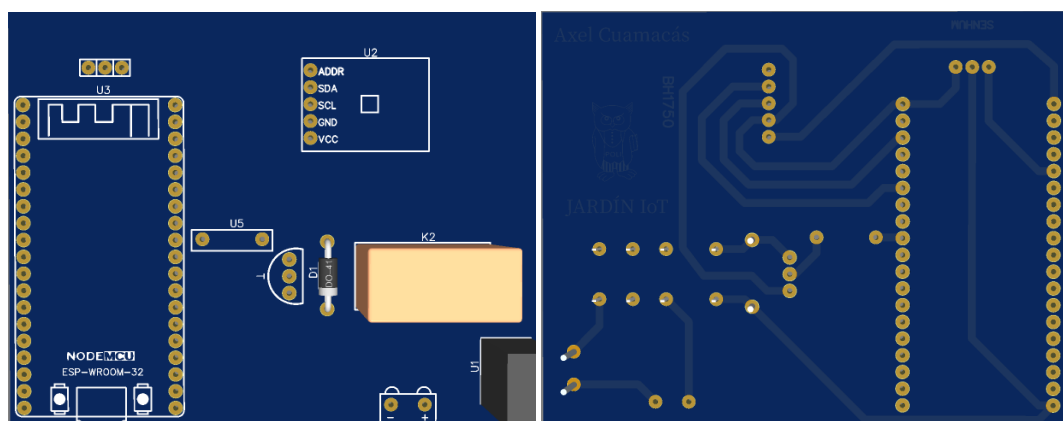


Figura 2.22 Visualización en 3D de la PCB en su capa inferior y superior

Posterior al diseño de la PCB se imprime el circuito en papel transferencia con una impresora láser y así utilizando el método del planchado, se transfiere el circuito a la lámina de fibra de vidrio FR4, se plancha el circuito por un periodo de entre 40 a 60 segundos

cuidando de que el papel transferencia no se mueva mientras es planchado. En la [Figura 2.23](#) se aprecia el resultado final luego de realizar este proceso.



Figura 2.23 Transferencia térmica del circuito a la placa de fibra de vidrio FR4

Se introduce la placa en ácido férrico agitando suavemente el envase en el que se encuentra para facilitar el proceso de quemado. En la [Figura 2.24](#) se ve el proceso y el resultado generado después de realizar este proceso.



Figura 2.24 Proceso de quemado de la PCB

Se realiza la protección del circuito impreso sumergiendo la placa en estaño líquido, aquí las pistas toman un tono plateado y se hacen las perforaciones necesarias para montar los elementos electrónicos sobre la placa, para esto se utiliza un dremel y una broca de 1 mm. El resultado se lo observa en la [Figura 2.25](#).



Figura 2.25 Protección del circuito impreso y perforación de los agujeros diseñados

Se montan los elementos electrónicos sobre la placa para ser soldado ([Figura 2.26](#)), considerando que para el ESP32 y los sensores se utiliza zócalos para tener la opción de remover estos elementos y utilizarlos en futuros proyectos. Además, se utilizan borneras para facilitar la conexión de la bomba de agua y el sistema de alimentación de la misma.

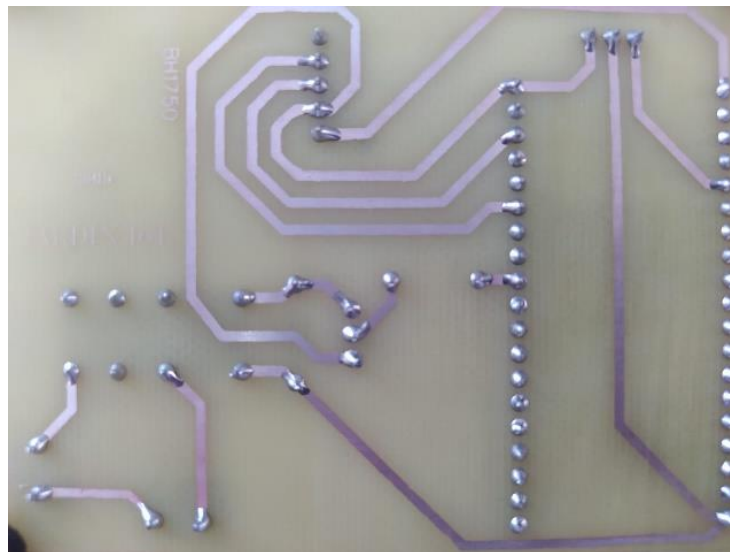


Figura 2.26 Soldado de los elementos en la PCB

Finalmente se sueldan los pines de cada elemento en la capa inferior de la placa como se observa en la [Figura 2.26](#), y se coloca el módulo ESP32 con los sensores según el diseño realizado. El resultado final de la PCB se lo observa en la [Figura 2.27](#).

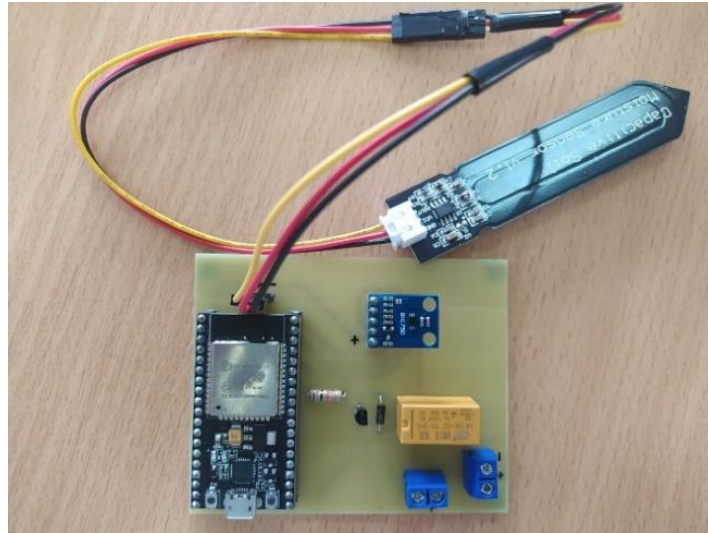


Figura 2.27 Resultado final de la PCB diseñada

2.5.2 Desarrollo del Módulo Protector

Dado que el prototipo pretende funcionar tanto para ambientes *indoor* como ambientes *outdoor*, se necesita de un módulo protector el cual se diseña utilizando la plataforma **Jerome Leary Box** para el diseño de la caja y **CoreIDRAW** para agregar los orificios al módulo protector que permita cargar el prototipo y la apertura para los sensores. El módulo está hecho de acrílico con un grosor de 3 mm y es cortado con una cortadora láser de alta precisión y el diseño realizado se lo muestra en la [Figura 2.28](#).

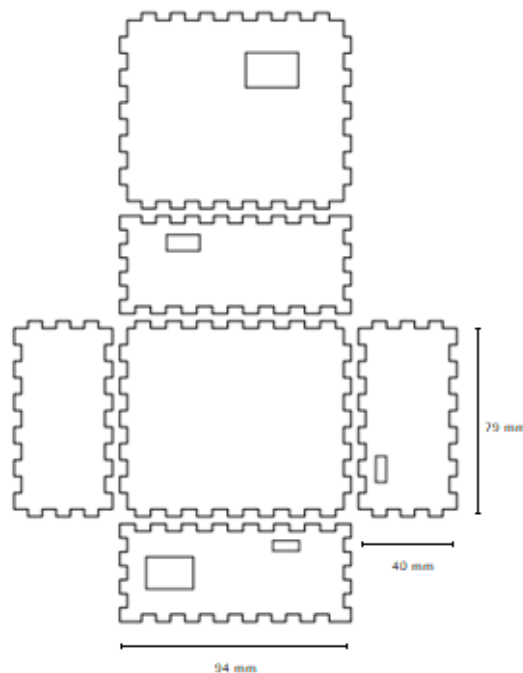


Figura 2.28 Diseño del módulo protector

Se introduce la placa dentro del módulo protector como se observa en la [Figura 2.29](#).

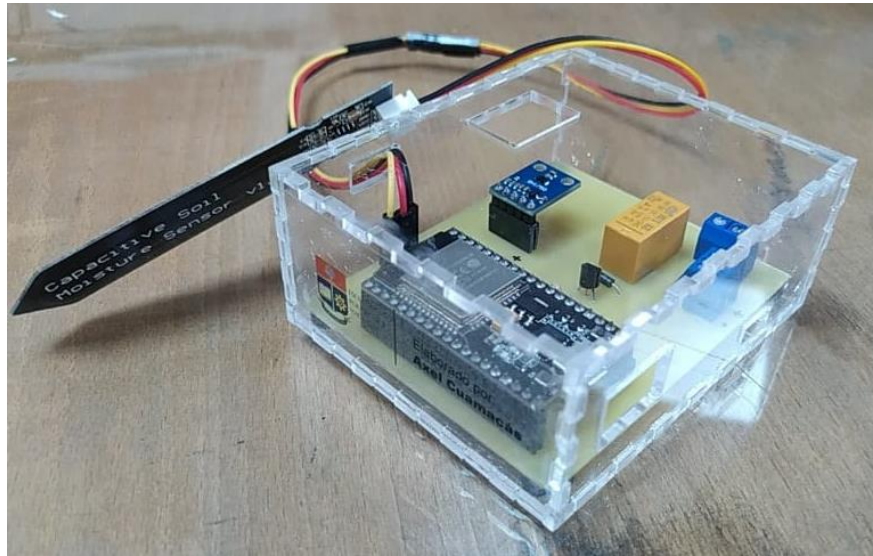


Figura 2.29 Prototipo final protegido por el modulo protector

3 PRUEBAS, RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Pruebas de Funcionamiento

En esta sección se establecen las pruebas de funcionamiento realizadas al prototipo, las cuales permiten determinar su correcto funcionamiento. En el [Anexo V](#) se presenta el enlace hacia las pruebas de funcionamiento realizadas de forma detallada.

3.1.1 Prueba de los Sensores

Se debe mencionar que previo a trabajar con el sensor capacitivo de humedad en el suelo v1.2 y con el sensor de luz digital BH1750, detallados en la sección [2.2.2](#) y [2.2.3](#) respectivamente, inicialmente se realizaron pruebas con el sensor resistivo FC-28 para la detección de humedad en el suelo, el cual fue descartado dado que los valores obtenidos por el sensor se ven afectados por el tipo de agua utilizada y se tarda en estabilizar la lectura de datos. Para la iluminancia se realizó pruebas con el sensor de luz LDR, el cual varia su resistencia dependiendo de la cantidad de luz recibida; sin embargo, este fue descartado dado que al realizar pruebas en ambientes con niveles elevados de iluminación, es decir con la luz del sol a medio día al aire libre, este se satura y se tienen niveles erróneos de iluminancia.

3.1.2 Recopilación de Información en una Base de Datos para Diferentes Ambientes de Prueba

Se implementa una configuración alterna en el ESP32 para realizar pruebas de funcionamiento en diferentes ambientes, realizando la lectura de humedad en el suelo e iluminancia por el método HTTP POST, el cual envía datos a un servidor web. Se realiza la conexión y el ingreso de datos por medio de una aplicación PHP alojada en el servidor, donde PHP es un lenguaje de programación que permite programar páginas con contenido dinámico, donde los datos son enviados en lenguaje HTML [59].

En la [Figura 3.1](#) se observa gráficamente la forma en la que se realiza el envío de datos a MySQL desde el ESP32 utilizando el método HTTP POST, para finalmente descargar estos datos y ser visualizados en Excel. El código generado en el IDE de Arduino para realizar este envío se encuentra en el [Anexo II](#).

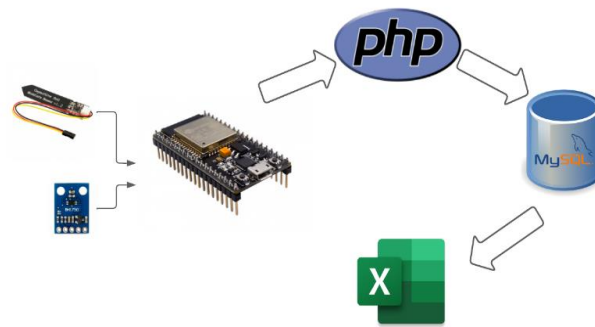


Figura 3.1 Envío de datos a MySQL desde el ESP32 utilizando el método HTTP POST

Se utiliza el host gratuito 000webhost para generar la ubicación de la aplicación php y así enviar los datos. Se debe considerar que el plan gratuito de 000webhost permite crear únicamente un sitio web con un almacenamiento máximo de 300 MB, por lo que solo se lo utiliza para pruebas más no constantemente en el prototipo [56]. Considerando que para ver los datos en tiempo real se usa el *dashboards* de Arduino IoT Cloud.

En 000webhost se selecciona la opción “Administrador de archivos”, donde se cargan los archivos en la raíz del sitio web y en la carpeta public.html se suben las aplicaciones php que permiten conectar con la base de datos, de igual forma estas aplicaciones php se visualizan en el [Anexo II](#). Por medio de phpMyAdmin se administra los valores de las variables con la fecha en la que se almacena los datos como se observa en la [Figura 3.2](#).

					Id	Iluminancia	Humedad	fecha_actual
<input type="checkbox"/>	Editar	Copiar	Borrar		174	9024	68	2023-02-07 11:56:27
<input type="checkbox"/>	Editar	Copiar	Borrar		175	8512	66	2023-02-07 11:57:01
<input type="checkbox"/>	Editar	Copiar	Borrar		176	9343	65	2023-02-07 11:57:33
<input type="checkbox"/>	Editar	Copiar	Borrar		177	12732	64	2023-02-07 11:58:06
<input type="checkbox"/>	Editar	Copiar	Borrar		178	25222	65	2023-02-07 11:58:38
<input type="checkbox"/>	Editar	Copiar	Borrar		179	17187	65	2023-02-07 11:59:11
<input type="checkbox"/>	Editar	Copiar	Borrar		180	11527	66	2023-02-07 11:59:44
<input type="checkbox"/>	Editar	Copiar	Borrar		181	8887	66	2023-02-07 12:00:16
<input type="checkbox"/>	Editar	Copiar	Borrar		182	10474	65	2023-02-07 12:00:48
<input type="checkbox"/>	Editar	Copiar	Borrar		183	10432	65	2023-02-07 12:01:21
<input type="checkbox"/>	Editar	Copiar	Borrar		184	10180	66	2023-02-07 12:01:53
<input type="checkbox"/>	Editar	Copiar	Borrar		185	9665	67	2023-02-07 12:02:27
<input type="checkbox"/>	Editar	Copiar	Borrar		186	15985	67	2023-02-07 12:02:59
<input type="checkbox"/>	Editar	Copiar	Borrar		187	15497	64	2023-02-07 12:04:46
<input type="checkbox"/>	Editar	Copiar	Borrar		188	12262	63	2023-02-07 12:05:19
<input type="checkbox"/>	Editar	Copiar	Borrar		189	10055	64	2023-02-07 12:05:51

Figura 3.2 Base de datos en phpMyAdmin

Todo este trabajo se lo realiza para obtener una base de datos apropiada para los diferentes escenarios de prueba que se realizaron y así obtener resultados adecuados. Realizando varias pruebas se determina que el tomar una muestra cada 30 segundos por 5 minutos es adecuado para realizar las pruebas en los diferentes escenarios (considerando que existe una demora de entre 2 y 3 segundos al conectarse con el servidor), se considera que en este tiempo se puede apreciar la variación que existe en la iluminancia para ambientes *outdoor*, mientras que para ambientes *indoor* se puede apreciar cómo la iluminancia generalmente tiende a ser constante.

3.1.3 Comparación entre los Valores obtenidos por el Sensor BH1750 y un luxómetro

Para validar los datos obtenidos por el sensor de luz digital BH1750 se realiza una comparación con los datos generados por aplicación Luxómetro para celular, la cual puede ser descargada en la Play Store [haciendo clic aquí](#).

Se debe considerar que para efectuar una toma de valores adecuados en la aplicación se debe calibrar el sensor de luz con un factor de 0.5 y así obtener valores similares a los otorgados por el sensor de luz digital BH1750; además, esta aplicación permite grabar los datos generados y ajustar el número de muestras deseadas para posteriormente ser exportados y visualizados en Excel. Para la prueba se toma 1 valor de iluminancia cada 30 segundos por 5 minutos.

En la [Tabla 3.1](#) se muestra los valores de Iluminancia obtenidos por la aplicación y el sensor de luz digital BH1750. Se debe de mencionar que estos datos fueron obtenidos entre las 14:08 horas y las 14:14 horas en un ambiente *indoor*.

Con la ayuda de la Ecuación 3.1 se puede obtener el error existente entre la iluminancia detectada por la aplicación y la detectada por el sensor BH1750.

$$\%Error_{ilum} = \frac{I_{lum_{app}} - I_{lum_{BH1750}}}{I_{lum_{BH1750}}} * 100$$

Ecuación 3.1

Donde:

$\%Error_{ilum}$: Error de iluminancia (%)

$I_{lum_{app}}$: Iluminancia obtenida por la aplicación (*lux*)

$I_{lum_{BH1750}}$: Iluminancia obtenida por el sensor BH1750 (*lux*)

Tabla 3.1 Valores de Iluminancia de la aplicación y del sensor BH1750

Iluminancia App (lux)	Iluminancia BH1750 (lux)	%Error	Hora de Muestra
288	295	2.3%	14:08:02
289	290	0.3%	14:08:35
288	290	0.6%	14:09:07
286	291	1.7%	14:10:55
287	283	1.4%	14:11:27
283	274	3.2%	14:12:00
256	266	3.7%	14:12:32
256	266	3.7%	14:13:05
256	268	4.4%	14:13:38
258	270	4.4%	14:14:11

En la [Figura 3.3](#) se muestra gráficamente cómo se comporta la iluminancia obtenida por la aplicación y por el sensor BH1750. Se observa claramente una similitud en su comportamiento.

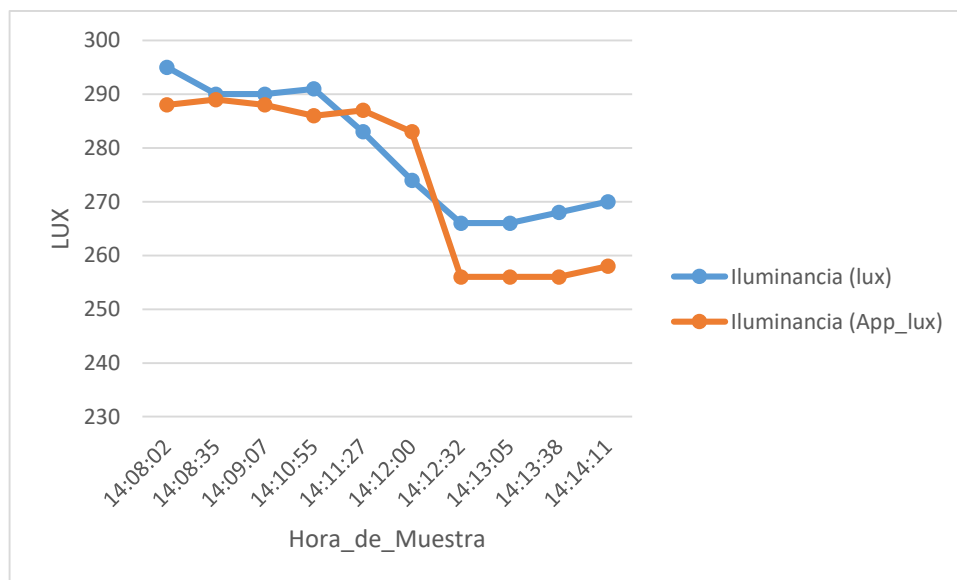


Figura 3.3 Grafica de la Iluminancia de la aplicación y del sensor BH1750

3.1.4 Nivel de Iluminancia con el Sensor de Luz Digital BH1750

Apoyándose en la [Tabla 1.7](#) de la [Sección 1.4.1.2](#) se considera que el nivel mínimo de iluminancia soportable para una planta es de 200 lux, dado que entre 0 y 200 lux se

menciona que es una intensidad de luz inadecuada para todas las plantas, así cuando la iluminancia sea inferior a 200 lux se envía un correo de alerta al usuario para que tome las precauciones del caso.

3.1.5 Nivel de Humedad en el Suelo obtenido con el Sensor Capacitivo de Humedad en el Suelo v1.2

Apoyándose en la [Tabla 1.6](#) de la [Sección 1.4.1.1](#) se considera un rango de humedad en el suelo ente el 21% y el 40%, que es el recomendable para las flores, por lo que cuando se tenga un nivel de humedad en el suelo menor al 21% la bomba de agua se encenderá automáticamente para dar riego a las plantas, y al llegar al 40% o superior se apagará automáticamente la bomba de agua, esto se puede observar en la programación realizada en el [Código 2.4](#).

Otro factor a tomar en cuenta es que al realizar las pruebas con el sensor capacitivo de humedad en el suelo v1.2, el ESP32 obtiene un valor de 4095 (0%) cuando el sensor se encuentra en el aire y 2000 (100%) cuando el sensor se encuentra totalmente sumergido en agua, esto se puede observar en la programación realizada en el [Código 2.2](#).

3.1.6 Prueba de Funcionamiento del envío de correos de alerta

Como se menciona en la [Sección 3.1.4](#) en el momento que la iluminancia es inferior a 200 lux se envía automáticamente un correo de alerta al usuario, el cual indica que se ha detectado niveles inadecuados en la intensidad de Luz sobre el jardín, recomendando verificar y corregir la intensidad de luz a la que está expuesta el jardín, esto se muestra en la [Figura 3.4](#).



Se a detectado niveles inadecuados en la intensidad de luz sobre el jardín

Webhooks via IFTTT <action@ifttt.com>
Responder a: Do not reply <no-reply@ifttt.com>
Para: axlbl12cuamacas@gmail.com

Se requiere verificar y corregir la intensidad de luz a la que esta expuesta el jardín.



Figura 3.4 Alerta generada vía Email

3.1.7 Prueba de Funcionamiento de la bomba de agua

Como se menciona en la [Sección 3.1.5](#) en el momento que la humedad sea inferior al 21% la bomba de agua se encenderá automáticamente para dar riego a las plantas y al llegar al 40% o superior se apagará automáticamente. Se realiza pruebas de funcionamiento de la bomba de agua dejando al aire libre el sensor manteniendo así encendida la bomba de agua hasta que el sensor sea nuevamente colocado en la planta, así las diversas pruebas realizadas fueron exitosas en diferentes ambientes y una de estas pruebas se la puede observar en la [Figura 3.5](#).



Figura 3.5 Activación automática de la bomba de agua

Se debe considerar que la bomba de agua tiene una fuente de alimentación externa con una pila recargable vista en la [Figura 2.19](#), se realiza pruebas de su durabilidad en la que se mantiene encendida la bomba de agua por un lapso aproximado de 4 horas consecutivas, donde se observó una reducción en su rendimiento, para lo cual se volvió a recargar la pila y el funcionamiento de la bomba de agua continuó correctamente.

3.1.8 Prototipo en Ambientes de Prueba Outdoor e Indoor

En esta sección se presentan los dos ambientes en los que se realizó pruebas, considerando diferente horario y clima.

- **Ambiente de Prueba *Outdoor***

Las pruebas en ambiente *Outdoor* se las realiza en un jardín de geranios como se ve en la [Figura 3.6](#), estas plantas son muy comunes en las casas de las personas sobre todo en zonas rurales.



Figura 3.6 Pruebas en un ambiente *Outdoor*

- **Ambiente de Prueba *Indoor***

Las pruebas en ambiente Indoor se las realiza en plantas ubicadas junto a una ventana como se ve en la [Figura 3.7](#).



Figura 3.7 Pruebas en un ambiente *Indoor*

- **Ambiente de Prueba *Indoor* con Luz Artificial**

Las pruebas en ambiente Indoor con luz artificial (luz led de 20W) se las realiza en una habitación cerrada sin luz solar entrante como se ve en la [Figura 3.8](#).



Figura 3.8 Pruebas en un ambiente *Indoor con luz artificial*

3.1.9 Pruebas de Visualización y Monitoreo de datos en Arduino IoT Cloud

Se realiza las pruebas de funcionamiento en el envío de datos hacia Arduino IoT Cloud, se puede visualizar los datos en computadora como se ve en la [Figura 3.9](#) y en el celular como se presenta en la [Figura 3.10](#). Los datos pueden ser monitoreados en tiempo real, en lapsos de 1 hora, 1 día, 7 días y hasta 15 días.

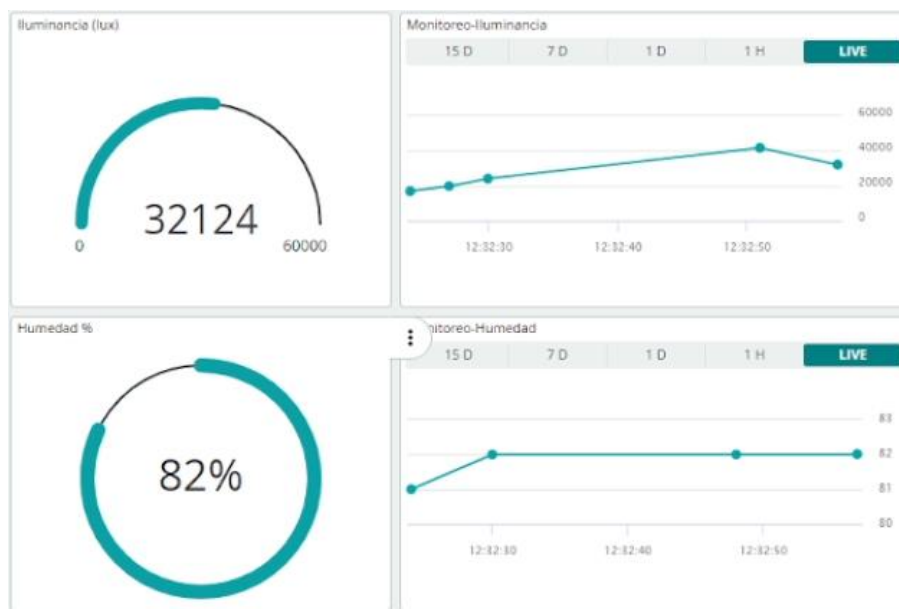


Figura 3.9 Visualización y Monitoreo de datos en Arduino IoT Cloud mediante la computadora

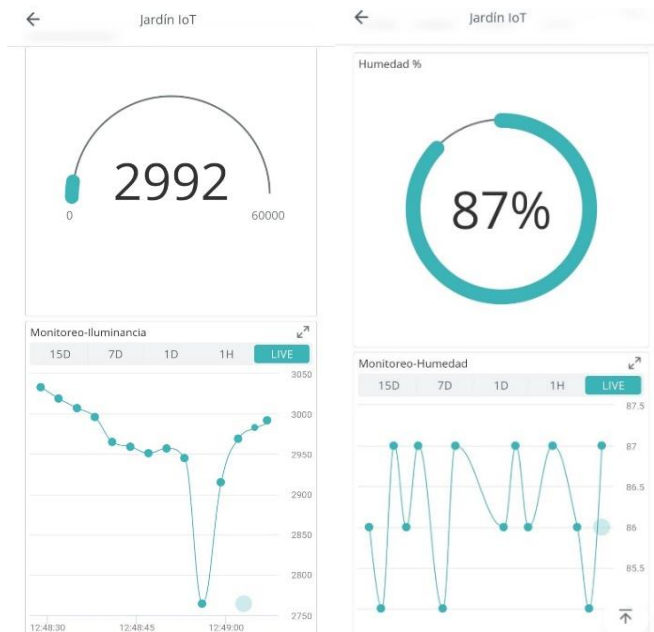


Figura 3.10 Visualización y Monitoreo de datos en Arduino IoT Cloud mediante el celular

3.2 Resultados

En esta sección se exponen los resultados generados en los diferentes escenarios de prueba. Se toma una muestra cada 30 segundos durante 5 minutos, considerando una demora de 2 a 3 segundos para conectarse con el servidor que guardara los datos.

- **Escenario 1:** Para este escenario se considera un ambiente *Outdoor* mayormente nublado. La toma de muestras inicia a las 11:59:44 y termina a las 12:05:51 como se observa en la [Tabla 3.2](#).

Tabla 3.2 Muestras de Iluminancia y Humedad en el Escenario 1

Iluminancia (lux)	Humedad (%)	Hora de Muestra
11527	66	11:59:44
8887	66	12:00:16
10474	65	12:00:48
10432	65	12:01:21
10180	66	12:01:53
9665	67	12:02:27
15985	67	12:02:59
15497	64	12:04:46
12262	63	12:05:19
10055	64	12:05:51

En la [Figura 3.11](#) se muestra como varía considerablemente la Iluminancia en el Escenario 1 mientras que la humedad en el suelo se mantiene constante.

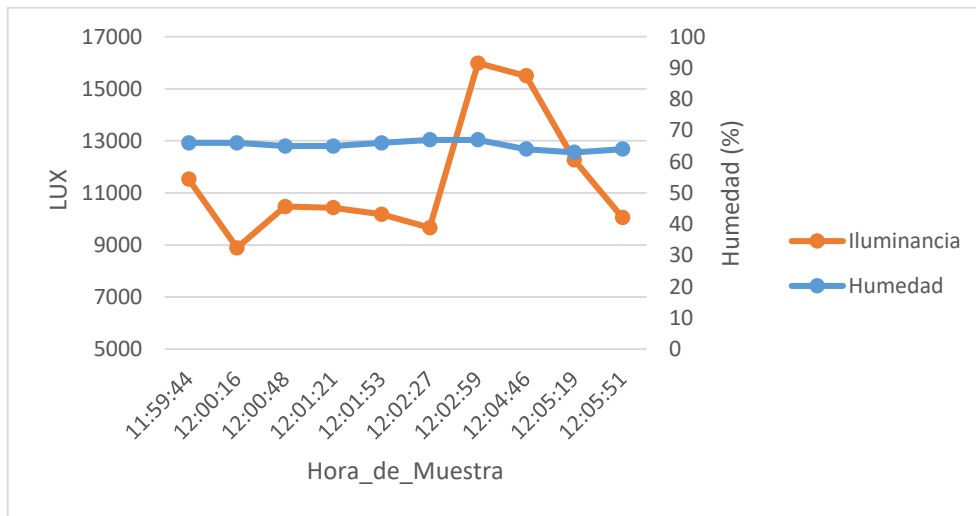


Figura 3.11 Gráfica de Iluminancia y Humedad en el Escenario 1

- **Escenario 2:** Para este escenario se considera un ambiente *Outdoor* soleado. La toma de muestras inicia a las 12:03:22 y termina a las 12:08:40 como se observa en la [Tabla 3.3](#).

Tabla 3.3 Muestras de Iluminancia y Humedad en el Escenario 2

Iluminancia (lux)	Humedad (%)	Hora de Muestra
54023	44	12:03:22
54612	44	12:03:54
50672	45	12:04:48
49033	46	12:05:26
47027	45	12:05:59
46036	45	12:06:33
57044	44	12:07:05
53028	45	12:07:38
52302	45	12:08:08
49701	44	12:08:40

En la [Figura 3.12](#) se muestra como la Iluminancia no varía tanto en el Escenario 2 y la humedad en el suelo se mantiene constante.

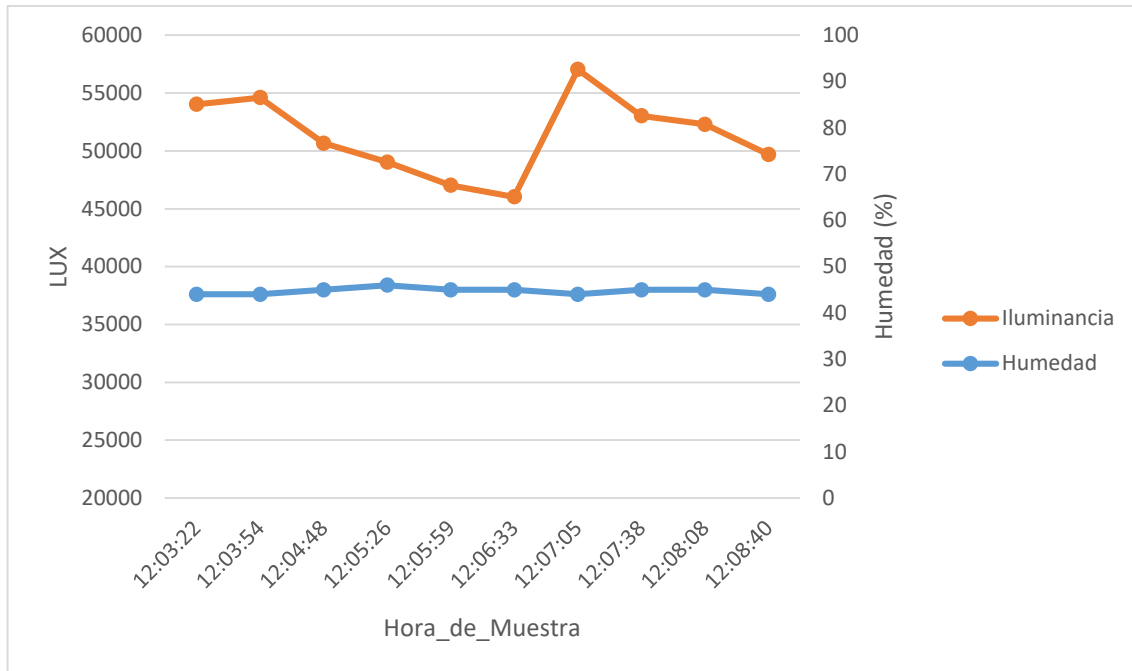


Figura 3.12 Gráfica de Iluminancia y Humedad en el Escenario 2

- **Escenario 3:** Para este escenario se considera un ambiente *Indoor* con leves niveles de luz solar entrante. La toma de muestras inicia a las 14:13:53 y termina a las 14:18:47 como se observa en la [Tabla 3.4](#).

Tabla 3.4 Muestras de Iluminancia y Humedad en el Escenario 3

Iluminancia (lux)	Humedad (%)	Hora de Muestra
454	76	14:13:53
408	76	14:14:26
336	73	14:14:58
309	75	14:15:31
300	76	14:16:04
298	76	14:16:37
325	76	14:17:10
369	76	14:17:42
406	75	14:18:15
421	76	14:18:47

En la [Figura 3.13](#) se muestra como la Iluminancia se ve reducida considerablemente para el Escenario 3 y la humedad en el suelo se mantiene constante.

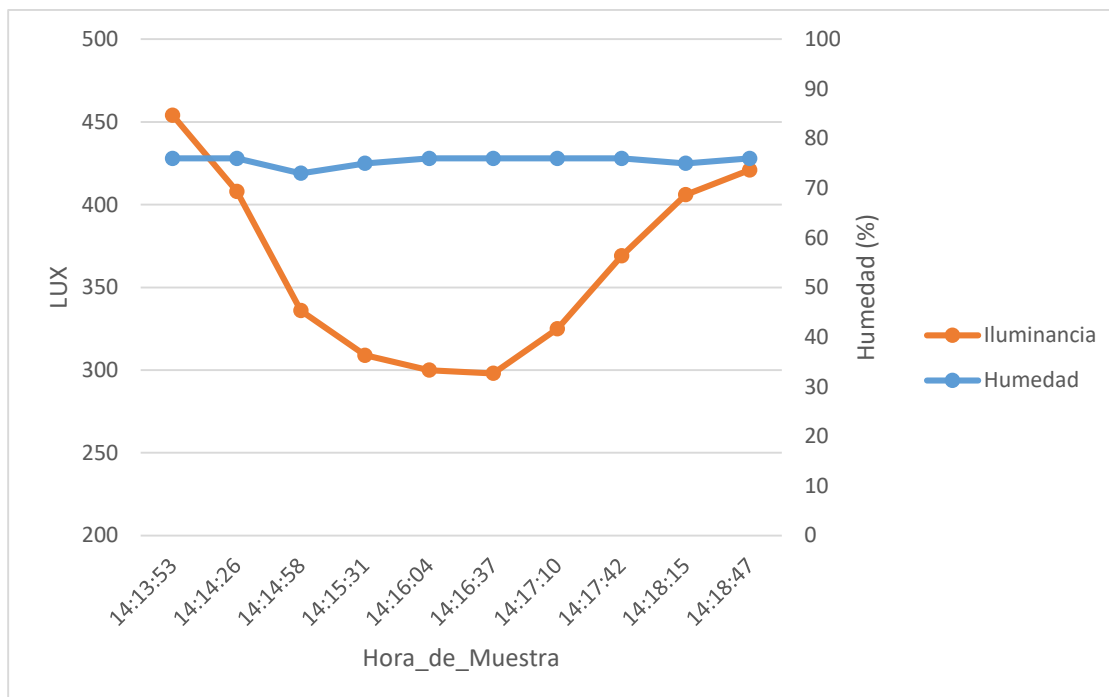


Figura 3.13 Grafica de Iluminancia y Humedad en el Escenario 3

- **Escenario 4:** Para este escenario se considera un ambiente *Indoor* con luz artificial en una habitación cerrada sin luz solar entrante. La toma de muestras inicia a las 19:08:14 y termina a las 19:13:08 como se observa en la [Tabla 3.5](#).

Tabla 3.5 Muestras de Iluminancia y Humedad en el Escenario 4

Iluminancia	Humedad	Hora de Muestra
1958	68	19:08:14
1958	68	19:08:46
1954	68	19:09:19
1954	68	19:09:52
1954	68	19:10:24
1954	68	19:10:57
1954	71	19:11:30
1953	68	19:12:02
1955	69	19:12:35
1954	68	19:13:08

En la [Figura 3.14](#) se muestra como la Iluminancia y la humedad en el suelo se mantienen constantes con una fuente de luz artificial.

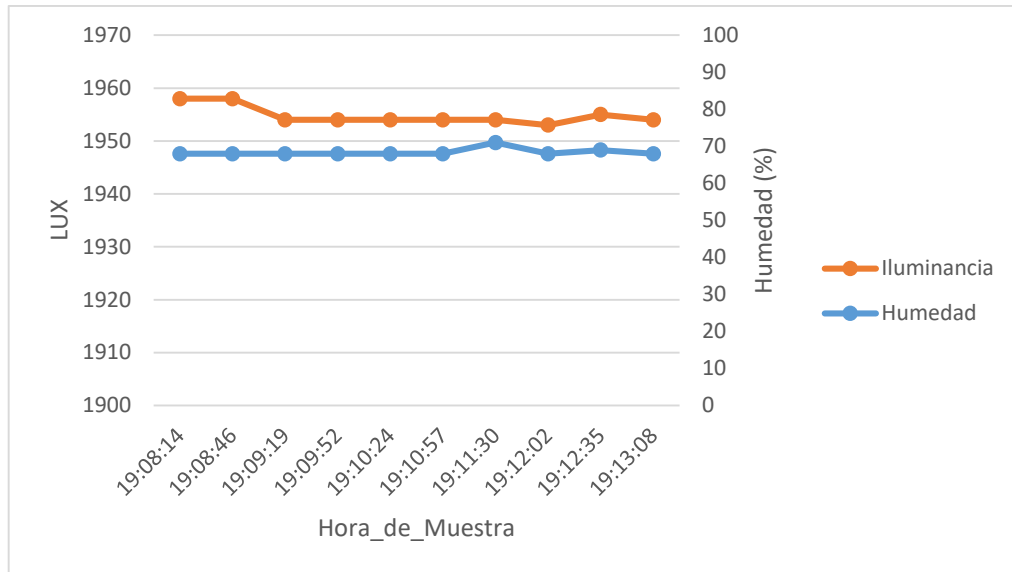


Figura 3.14 Gráfica de Iluminancia y Humedad en el Escenario 4

3.3 Conclusiones

- El prototipo implementado permite la detección del nivel de humedad del suelo en porcentaje mediante el sensor capacitivo de humedad en el suelo v1.2, y la iluminación mediante el sensor de luz digital BH1750 en luxes, se monitorea en tiempo real los datos generados en la plataforma de Arduino IoT Cloud, se genera una alerta vía email cuando el nivel de iluminación inadecuado y se activa un actuador automáticamente cuando el porcentaje de humedad en el suelo es deficiente.
- El módulo Wi-Fi ESP32 es una excelente opción para la conectividad inalámbrica a IoT por Wi-Fi, este permite el envío de datos hacia un servidor en la nube y poder gestionar estos datos según las necesidades del usuario; además, el tamaño del ESP32 y la facilidad con la que se puede programar es muy atractiva para realizar proyectos IoT, contando con los pines programables para satisfacer las necesidades del prototipo.
- La compatibilidad del módulo Wi-Fi ESP32 con el IDE de Arduino es una ventaja significativa para los nuevos desarrolladores que utilizan este módulo por primera vez. El IDE de Arduino es muy amigable para el usuario y su entorno está diseñado para simplificar la tarea de programar y grabar códigos.
- Para el diseño de la PCB el utilizar la plataforma EasyEDA fue una opción acertada, dado que no se requiere descargar e instalar nada, solo se requiere de crear una cuenta para continuar trabajando sobre esta plataforma, además de que se ofrece

una amplia gama de herramientas para el diseño de circuitos electrónicos y como estos se pueden trasladar de forma sencilla a placas de circuito impreso.

- El implementar la PCB y el modulo protector al circuito realizado para el prototipo es de vital importancia para poder realizar las pruebas de funcionamiento en ambientes que podrían ser hostiles, manejando de mejor manera los componentes electrónicos implementados en el prototipo.
- Revisando los resultados obtenidos se verifica que en ambientes *Indoor* los niveles de iluminancia son constantes, mientras que los niveles de iluminancia en ambientes *Outdoor* varían constantemente, esto debido a las distintas condiciones ambientales que se puede tener, por otro lado, los niveles de humedad en el suelo permanecen más estables y estos varían en un lapso más prolongado de tiempo.
- El prototipo implementado permite reducir tiempo, recursos y errores humanos, controlando constantemente el estado en el que se encuentran las plantas, y así tener una visión más adecuada de como las condiciones ambientales en las que se encuentran las plantas afectan directamente a su crecimiento y floración.

3.4 Recomendaciones

- Con el fin de evitar daños en el módulo Wi-Fi ESP32 se recomienda alimentar la bomba de agua con una fuente de alimentación externa, lo que permite a su vez poder cambiar a una bomba de agua de mayor potencia, siendo esto recomendable para poder dar riego a una mayor extensión de terreno en el jardín.
- Es recomendable utilizar un sensor capacitivo en lugar de un resistivo para la detección de humedad en el suelo, dado que los sensores capacitivos son más estables y precisos en sus mediciones, posee una mejor respuesta a los cambios de humedad y no son tan susceptibles a la corrosión.
- Se recomienda utilizar el sensor BH1750 para medir la iluminación en lugar de un módulo LDR, dado que se tiene una mayor precisión al tener una respuesta espectral más uniforme, por lo que se tiene una mayor sensibilidad a diferentes longitudes de onda y es menos susceptible a variaciones de temperatura.
- En cuanto al nivel de iluminación que se tiene en ambientes oscuros se recomienda utilizar una fuente de luz artificial. Para las pruebas se utiliza una luz led de 20W, la cual da 1954 luxes aproximadamente como se observa en los resultados, pero esta iluminancia cubre únicamente una porción pequeña de terreno en un jardín, por lo

que se requiere una luz de mayor potencia o de un mayor número de fuentes de luz que cubran la totalidad del jardín.

- Se recomienda utilizar una fuente de alimentación diferente al cargador común de celular para alimentar el ESP32, esta podría ser una batería recargable con un regulador de voltaje, permitiendo de este modo moverse de mejor forma con el prototipo en los diferentes lugares del jardín y no estar limitados por el largo del cable del cargador.
- Es recomendable impermeabilizar el módulo protector del prototipo, evitando los daños que puedan generarse por agua ya sea en el riego de las plantas o en lluvias.

4 REFERENCIAS BIBLIOGRÁFICAS

- [1] Lifelong Learning, "Ingeniería de los sistemas embebidos.," *Ind. Syst. engineering*, pp. 1–19, 2011, [Online]. Available: http://www.ieec.uned.es/investigacion/dipseil/pac/archivos/informacion_de_referencia_ise5_3_1.pdf (accessed Nov. 19, 2022).
- [2] J. E. Bonells, "LA TERAPIA HORTÍCOLA Y LOS JARDINES TERAPÉUTICOS," 2020. <https://jardinessinfronteras.com/2020/08/24/la-terapia-horticola-y-los-jardines-terapeuticos/> (accessed Nov. 19, 2022).
- [3] J. Perea, "Diseño de un sistema de monitoreo, registro y control de temperatura y humedad para un cultivo de invernadero.," *Univ. Tecnológica Pereira*, pp. 1–79, 2016, [Online]. Available: <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/7436/621317P434.pdf?sequence=1> (accessed Nov. 20, 2022).
- [4] V. Puranik, Sharmila, A. Ranjan, and A. Kumari, "Automation in Agriculture and IoT," *Proc. - 2019 4th Int. Conf. Internet Things Smart Innov. Usages, IoT-SIU 2019*, pp. 1–5, 2019, doi: 10.1109/IoT-SIU.2019.8777619. (accessed Nov. 22, 2022).
- [5] INTAGRI, "Efectos del Exceso de Humedad del Suelo en el Sistema Radical y Absorción de Nitrógeno en el Maíz." <https://www.intagri.com/articulos/agua-riego/exceso-humedad-del-suelo-en-sistema-radical> (accessed Nov. 18, 2022).
- [6] C. Beaza, "LA LUZ COMO FACTOR AMBIENTAL PARA LAS PLANTAS," 2020. <https://nutricontrol.com/es/politica-de-privacidad/> (accessed Nov. 18, 2022).
- [7] V. Cherlinka, "I Control De La Humedad Del Suelo: Un Factor Clave," 2020. <https://eos.com/es/blog/humedad-del-suelo/#:~:text=El potencial de humedad del,la tierra%2C más agua absorberá> (accessed Nov. 15, 2022).
- [8] J. Enciso, D. Porter, and X. Péries, "Uso de sensores de humedad en suelo para mejorar el manejo del riego en cultivos de invernadero," *Mejor. en la Efic. del uso agua y Fertil. en Agric.*, vol. 39, no. 5, pp. 107–132, 2018, [Online]. Available: <http://riograndewater.org/media/1080/e-618s-irrigation-monitoring-with-soil-water-sensors-spanish-version.pdf> (accessed Nov. 16, 2022).
- [9] Tecnoriegovalley, "DETERMINACIÓN DEL CONTENIDO HÍDRICO DEL SUELO", [Online]. Available: <https://www.tecnoriegovalley.com.ar/uploads/fca-unc-determinacion-del-contenido-hidrico-del-suelo-2983.pdf> (accessed Nov. 27, 2022).

- [10] Diferenciador, “Tipos de suelos.” <https://www.diferenciador.com/tipos-de-suelos/> (accessed Nov. 18, 2022).
- [11] ACURITE, “Soil Moisture Recommendations for Flowers, Plants, and Vegetables,” 2019. <https://www.acurite.com/blog/soil-moisture-guide-for-plants-and-vegetables.html#:~:text=Recommended Soil Moisture Levels&text=It is important to note,between 41%25 - 80%25> (accessed Feb. 03, 2023).
- [12] S. B. Agrawal, S. Singh, and M. Agrawal, “Role of Light in Plant Development,” *Int. J. Plant Environ.*, vol. 1, no. 01, pp. 43–56, 2015, doi: 10.18811/ijpen.v1i1.7113. (accessed Feb. 04, 2023).
- [13] Hydroenv, “La luz en tus plantas.” https://www.hydroenv.com.mx/catalogo/index.php?main_page=page&id=221 (accessed Nov. 26, 2022).
- [14] Sunday Gardener, “How Many LUX for Indoor Plants,” 2022. <https://www.sundaygardener.net/how-many-lux-for-indoor-plants/> (accessed Feb. 04, 2023).
- [15] C. Merino, “Colegio de Ciencias Sociales y Humanidades Uso de la Terapia Hortícola como estrategia terapéutica para reducir los síntomas del duelo complicado persistente en niños de 3 a 7 años y promover la adaptación,” 2019, [Online]. Available: <https://repositorio.usfq.edu.ec/bitstream/23000/8206/1/142597.pdf> (accessed Feb. 06, 2023).
- [16] O. Herrero, S. Tutora, and J. R. Mart, “TRABAJO DE FIN DE GRADO Efectos de la terapia hortícola en la salud física, mental y social de las personas. Una revisión narrativa.”, 2018, [Online]. Available: https://repositorio.uam.es/bitstream/handle/10486/684889/herrero_solana_victor_omartfg.pdf?sequence=1&isAllowed=y (accessed Feb. 08, 2023).
- [17] A. Samuel and C. Sipes, “Making Internet of Things Real,” *IEEE Internet Things Mag.*, vol. 2, no. 1, pp. 10–12, 2019, doi: 10.1109/iotm.2019.1907777. (accessed Feb. 08, 2023).
- [18] L. K. Ramasamy and S. Kadry, “Internet of things (IoT),” *Blockchain Ind. Internet Things*, no. May, 2021, doi: 10.1088/978-0-7503-3663-5ch1. (accessed Feb. 17, 2023).
- [19] J. Fernández, “IoT en América Latina en 2023 habrá 996 millones de dispositivos

- conectados,” *America Retail*, 2022. <https://www.america-retail.com/innovacion/iot-en-america-latina-en-2023-habra-996-millones-de-dispositivos-conectados/> (accessed Oct. 26, 2022).
- [20] A. De Sistemas and B. Ú. Miñarro, “Sistemas embebidos,” *Univ. Murcia*, pp. 1–30, 2009, [Online]. Available: <https://www.um.es/documents/4874468/19345367/ssee-t02.pdf/bdf91c12-df9a-4b80-9e15-2241ee6eabad> (accessed Feb. 04, 2023).
- [21] N. Campana, “¿Qué hace un desarrollador de sistemas embebidos?,” *Freelancermap*, 2020. <https://www.freelancermap.com/blog/es/que-hace-desarrollador-sistemas-embebidos/> (accessed Oct. 26, 2022).
- [22] Tableroalparque, “Introduccion a los sensores,” pp. 33–60, 2021, [Online]. Available: https://tableroalparque.weebly.com/uploads/5/1/6/9/51696511/introducciÓn_a_los_sensores.pdf (accessed Feb. 05, 2023).
- [23] J. F. Vetelino and A. Reghu, “Introduction to sensors,” *Introd. to Sensors*, no. November, pp. 1–180, 2017, doi: 10.1201/9781315218274. (accessed Feb. 03, 2023).
- [24] P. Marek, “SENSORS : Types and Characteristics I- Physical Measurements as input,” 2019, [Online]. Available: <https://web.iit.edu/sites/web/files/departments/academic-affairs/academic-resource-center/pdfs/SENSORS.pdf> (accessed Feb. 04, 2023).
- [25] J. Miller and D. Bitecofer, “Soil Moisture Sensor Soil Moisture Sensor Final Design Report,” pp. 3–105, 2018, [Online]. Available: <https://core.ac.uk/download/pdf/232684239.pdf> (accessed Feb. 01, 2023).
- [26] T. A. Papalias and M. Wong, “Making sense of light sensors,” *Electron. Eng. Times*, no. 1429, 2006, [Online]. Available: https://www.researchgate.net/publication/298589142_Making_sense_of_light_sensors (accessed Feb. 03, 2023).
- [27] MecatronicaLatam, “LDR o fotoresistor o fotoresistencia,” 2021. <https://www.mecatronicalatam.com/es/tutoriales/sensores/sensor-de-luz/ldr/> (accessed Nov. 13, 2022).
- [28] MecatronicaLATAM, “Fotodiodo,” 2021. <https://www.mecatronicalatam.com/es/tutoriales/electronica/componentes-electronicos/diodo/fotodiodo/> (accessed Nov. 13, 2022).

- [29] TallerElectronica, "FOTOTRANSISTOR." <https://tallerelectronica.com/fototransistor/> (accessed Nov. 13, 2022).
- [30] RedWeb, "Los amplificadores de transimpedancia para fotodiodos de amplio rango presentan exigentes requisitos," 2015. <https://www.redeweb.com/articulos/instrumentacion/los-amplificadores-de-transimpedancia-para-fotodiodos-de-amplio-rango-presentan-exigentes-requisitos/#:~:text=> (accessed Nov. 13, 2022).
- [31] L. del Valle, "Proyectos IoT con Arduino, las plataformas más importantes," 2017. <https://programarfacil.com/podcast/proyectos-iot-con-arduino/> (accessed Nov. 17, 2022).
- [32] P. Bayborodin, "Features of the new Blynk IoT Platform," 2021. <https://blynk.io/blog/new-blynk-iot-platform-beta> (accessed Nov. 17, 2022).
- [33] Digitaltalent, "Metodologías de gestión de proyectos," pp. 1–16, 2018, [Online]. Available: https://www.dtagency.tech/cursos/metodologias_gestion_proyectos/tema_1-ModeloWaterfall.pdf (accessed Nov. 18, 2022).
- [34] ESP32Series, "ESP32 Series Datasheet," *Espr. Syst.*, pp. 1–65, 2021, [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf (accessed Nov. 19, 2022).
- [35] J. Guerra, "Programar ESP32 con Arduino IDE," 2022. <https://programarfacil.com/esp8266/programar-esp32-ide-arduino/#:~:text=ESP32 es una serie de,-Fi%2C también soporta Bluetooth.> (accessed Nov. 08, 2022).
- [36] V. Asanza, "Especificaciones del módulo ESP32." <https://vasanza.blogspot.com/2021/07/especificaciones-del-modulo-esp32.html> (accessed Jan. 10, 2023).
- [37] Soloarduino, "¿Qué es un ESP32?," 2017. <https://soloarduino.blogspot.com/2017/03/que-es-un-esp32.html> (accessed Nov. 18, 2022).
- [38] DFRobot, "Capacitive Soil Moisture Sensor," pp. 1–6, 2018, [Online]. Available: <https://www.sigmaelectronica.net/wp-content/uploads/2018/04/sen0193-humedad-de-suelos.pdf> (accessed Jan. 21, 2023).

- [39] Szhaiwang, “Módulo sensor capacitivo de humedad del suelo no fácil de corroer cable de amplio voltaje 3,3 ~ 5,5 V resistente a la corrosión con gravedad para Arduino.” <https://szhaiwang.en.made-in-china.com/product/LdpxOHQbbtVB/China-Capacitive-Soil-Moisture-Sensor-Module-Not-Easy-to-Corrode-Wide-Voltage-Wire-3-3-5-5V-Corrosion-Resistant-W-Gravity-for-Arduino.html> (accessed Jan. 29, 2023).
- [40] ROHM semiconductor, “Digital 16bit Serial Output Type Ambient Light Sensor IC,” no. 11046, p. 21, 2011, [Online]. Available: <https://www.mouser.com/datasheet/2/348/bh1721fvc-e-1809005.pdf> (accessed Jan. 14, 2023).
- [41] NAYLAMP, “Módulo sensor de luz digital BH1750.” <https://naylampmechatronics.com/sensores-luz-y-sonido/76-modulo-sensor-de-luz-digital-bh1750.html#:~:text=> (accessed Jan. 27, 2023).
- [42] C. Resistance, “HK19F HK19F”, [Online]. Available: https://www.lcsc.com/product-detail/Power-Relays_Ningbo-Keke-New-Era-Appliance-HK19F-DC5V-SHG-Y_C54512.html?utm_source=digipart&utm_medium=cpc&utm_campaign=HK19F-DC5V-SHG (Y%29 (accessed Jan. 14, 2023).
- [43] GROBOTRONICS, “Relay 2P 5V 1A Subminiature HK19F.” <https://grobotronics.com/2p-5v-1-subminiature-hk19f.html?sl=en> (accessed Feb. 13, 2023).
- [44] V. Dc and T. Si, “Para proyectos de riego automático Especificaciones :,” pp. 1–13, [Online]. Available: <https://www.bolanosdj.com.ar/MOVIL/ARDUINO2/bombadeagua.pdf> (accessed Jan. 17, 2023).
- [45] Patagoniatec, “Mini Bomba De Agua Sumergible.” <https://compras.patagoniatec.com/productos/mini-bomba-de-agua-sumergible-autocebante-2l-m-3v-5v-dc/> (accessed Jan. 14, 2023).
- [46] Karl Söderby, “Getting Started With the Arduino IoT Cloud,” 2022. <https://docs.arduino.cc/arduino-cloud/getting-started/iot-cloud-getting-started> (accessed Jan. 05, 2023).
- [47] IFTTT, “What is IFTTT?” https://ifttt.com/explore/new_to_ifttt (accessed Jan. 14, 2023).
- [48] R. García, “Qué es IFTTT y cómo te ayuda a que todo sea más fácil,” 2022.

- <https://www.adslzone.net/esenciales/preguntas/que-es-ifttt/> (accessed Jan. 14, 2023). (accessed Jan. 18, 2023).
- [49] Lamiacasaelettrica, "IFTTT italiano: qué es, cómo configurar para domótica," 2020. <https://lamiacasaelettrica.com/ifttt-italiano-come-funziona/> (accessed Jan. 20, 2023).
- [50] U. Date and E. Version, "EasyEDA Tutorial Introduction to EasyEDA What ' s EasyEDA," 2017, [Online]. Available: https://easyeda.com/Doc/Tutorial/Doc/EasyEDA_Tutorial_20170920.pdf (accessed Jan. 15, 2023).
- [51] INVENTABLE, "Como controlar un relé con un transistor." <https://www.inventable.eu/controlar-rele-con-transistor/> (accessed Jan. 16, 2023).
- [52] D. Sheet and M. Dimensions, "Description NPN General Purpose Transistor Data Sheet 2N2222A NPN General Purpose Transistor", [Online]. Available: <https://pdf1.alldatasheet.com/datasheet-pdf/view/956542/FCI/2N2222A.html> (accessed Jan. 16, 2023).
- [53] TECmikro, "2N2222 TRANSISTOR NPN." <https://tecmikro.com/transistores/16-2n2222-transistor-npn.html> (accessed Jan. 17, 2023).
- [54] Cytron, "Diodes 1N4007." <https://www.cytron.io/c-electronic-components/p-diodes-1n4007> (accessed Jan. 17, 2023).
- [55] INVENTABLE, "Introducción a los relés." <https://www.inventable.eu/introduccion-a-los-reles/> (accessed Jan. 17, 2023).
- [56] M. Šeikytė, "000WebHost review," 2022. <https://cybernews.com/best-web-hosting/000webhost-review/#:~:text=000WebHost is among the top,recommend it for professional use.> (accessed Jan. 27, 2023).
- [57] K. Söderby, "Connecting ESP32 & ESP8266 to Arduino Cloud IoT," 2022. <https://docs.arduino.cc/arduino-cloud/getting-started/esp-32-cloud> (accessed Jan. 11, 2023).

5 ANEXOS

ANEXO I. Código Fuente Principal

ANEXO II. Códigos Generador de Base de Datos Para Pruebas en Diferentes Ambientes

ANEXO III. Conexión del ESP32 con la Plataforma de Arduino Cloud IoT

ANEXO IV. Envío de Alerta por Email a través de la Plataforma IFTTT

ANEXO V. Enlace de pruebas de funcionamiento

ANEXO I. Código Fuente Principal

Código I.1 Código Fuente Principal (Parte 1)

```
//Escuela Politécnica Nacional
//Elaborado por: Axel Cuamacás
//Librerías
#include <ArduinoIoTCloud.h>
#include <Arduino_ConnectionHandler.h>
#include <Arduino.h>
#include <Wire.h>
#include <BH1750.h>
#include <WiFi.h>
#include <WiFiMulti.h>
#include <HTTPClient.h>
WiFiMulti wifiMulti;
const char DEVICE_LOGIN_NAME[] = "8e726d43-5345-49ed-b6e4-fb7f11eedfe5";
// Nombre del inicio de sesión de del dispositivo (generado
// automáticamente)
const char SSID[] = "NETLIFE-ESTHELA"; // Red SSID
(nombre)
const char PASS[] = "Martin1080"; // Contraseña de red
(usar para WPA o usar como clave para WEP)
const char DEVICE_KEY[] = "PL06CESKFOAASKVXNXJQ"; // Contraseña
secreta del dispositivo ESP32
int humedad;
int luminosidad;
void initProperties() {
    ArduinoCloud.setBoardId(DEVICE_LOGIN_NAME);
    ArduinoCloud.setSecretDeviceKey(DEVICE_KEY);
    ArduinoCloud.addProperty(humedad, READ, ON_CHANGE, NULL);
    ArduinoCloud.addProperty(luminosidad, READ, ON_CHANGE, NULL);
}
WiFiConnectionHandler ArduinoIoTPreferredConnection(SSID, PASS);
BH1750 lightMeter;
int SenCap = 34;
int HUM;
int ValHum;
const int ValAire = 4095; //Valor obtenido por el sensor de humedad en
el suelo en el aire
```

Código I.1 Código Fuente Principal (Parte 2)

```
const int ValAgua = 2000; //Valor obtenido por el sensor de humedad en
el suelo en el agua
String Url_LUZ =
"https://maker.ifttt.com/trigger/Sensor_iluminancia/json/with/key/j4ohQx8
yhUJU5jJeldbPMeFlR_gaU3M-XOvsImM3yeW";
void setup() {
  pinMode(18, OUTPUT); // pin donde se conectó el gatillo del relé
  Serial.begin(9600);
  Serial.println();
  Serial.println();
  Serial.println();
  for(uint8_t t = 4; t > 0; t--) {
    Serial.printf("CARGANDO %d...\n", t);
    Serial.flush();
    delay(1000);
  }
  wifiMulti.addAP(SSID, PASS);
  // Se inicializa el bus I2C
  Wire.begin();
  lightMeter.begin();
  Serial.println(F("Prueba bh1750 inicializada"));
  // Se llama a la función thingProperties.h
  initProperties();
  // Conectando a Arduino IoT Cloud
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  Serial.println("Ingresando a la lectura de los sensores de humedad e
iluminancia...");
  //Se obtiene mayor información relacionada con la conexión de IoT
Cloud, el estado de la red y los errores que puedan ocurrir
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
}
void loop() {
  //COMUNICACIÓN CON Arduino IoT cLOUD
  ArduinoCloud.update();
  //Valores finales de humedad
  HUM = analogRead(SenCap);
  ValHum = map(HUM, ValAgua, ValAire, 100, 0);

  //Valores finales de Iluminancia
  luminosidad = lightMeter.readLightLevel();
  //IMPRESIÓN DE DATOS
  if(ValHum >= 100)
  {
    humedad = 100;
    Serial.print("Humedad del suelo:
");Serial.print(humedad);Serial.println(" %");
  }
  else if(ValHum <=0)
  {
    humedad = 0;
    Serial.print("Humedad del suelo:
");Serial.print(humedad);Serial.println(" %");
  }
}
```

Código I.1 Código Fuente Principal (Parte 3)

```
}
else if(ValHum >0 && ValHum < 100)
{
    humedad = ValHum;
    Serial.print("Humedad del suelo:
");Serial.print(humedad);Serial.println(" %");
}
Serial.print("Iluminancia: ");Serial.print(luminosidad);Serial.println("
lux");
delay(2000);//pausa de 2 segundo solo para ver de mejor forma los valores
expuestos en el monitor serial
//CORREO DE ALERTA
if (luminosidad<200) // Si la variable luminosidad es menor que 200 lux
se procede a realizar la peticion HTTP
{
    if((wifiMulti.run() == WL_CONNECTED)) {
        HTTPClient http;
        Serial.print("Iniciando el envio de [HTTP] ...\\n");
        http.begin(Url_LUZ); //Envia la peticion
        Serial.print("Esperando respuesta de [HTTP] ...\\n");
        int httpCode = http.GET();

        if(httpCode > 0) {

            Serial.printf("Captura codigo [HTTP] ...: %d\\n", httpCode);
            if(httpCode == HTTP_CODE_OK) {
                String payload = http.getString();
                Serial.println(payload);
            }
        } else {
            Serial.printf("Error: Captura fallida [HTTP] GET... %s\\n",
http.errorToString(httpCode).c_str());
        }

        http.end();
        delay(30000); // El correo se envia cada 30 segundos mientras
se siga teniendo valores inadecuados de iluminancia
    }
}
//ACTUADOR PARA HUMEDAD EN EL SUELO
if (humedad<21) // Nivel de encendido de la bomba de agua
{
    Serial.println("Suelo seco - Bomba encendida");
    digitalWrite(18, HIGH); // Si se tiene una señal en alto en el relé
se enciende la bomba de agua
}
else if (humedad >= 40) // el nivel máximo de agua debe ser
{
    Serial.println("Suelo mojado, Bomba apagada");
    digitalWrite(18, LOW); // Si se tiene una señal en bajo en el relé se
apaga la bomba de agua
}
delay(1000);
```

ANEXO II. Códigos Generador de Base de Datos Para Pruebas en Diferentes Ambientes

Código II.1 Código generado en el IDE de Arduino para realizar el envío de datos a MySQL utilizando el método HTTP POST (Parte 1)

```
//Librerias
#include <Arduino_ConnectionHandler.h>
#include <Arduino.h>
#include <WiFi.h>
#include <WiFiMulti.h>
#include <HTTPClient.h>
#include <Wire.h>
#include <BH1750.h>
#include <WiFiClient.h>
WiFiClient client;
WiFiMulti wifiMulti;

const char SSID[]           = "Nombre_red";    // Red SSID (nombre)
const char PASS[]          = "Clave_red";     // Contraseña de red (usar
para WPA o usar como clave para WEP)
int humedad;
int luminosidad;
WiFiConnectionHandler ArduinoIoTPreferredConnection(SSID, PASS);
BH1750 lightMeter;
int SenCap = 34;
int HUM;
int ValHum;

const int ValAire = 4095; //Valor obtenido por el sensor de humedad en el
suelo en el aire
const int ValAgua = 2000; //Valor obtenido por el sensor de humedad en el
suelo en el agua
void setup() {

    Serial.begin(9600);
    Serial.println();
    Serial.println();
    Serial.println();

    for(uint8_t t = 4; t > 0; t--) {
        Serial.printf("ESPERE %d...\n", t);
        Serial.flush();
        delay(1000);
    }

    wifiMulti.addAP(SSID, PASS);

    // Se inicializa el bus I2C
    Wire.begin();
    lightMeter.begin();
    Serial.println(F("Prueba bh1750 inicializada"));
```

Código II.1 Código generado en el IDE de Arduino para realizar él envío de datos a MySQL utilizando el método HTTP POST (Parte 2)

```
//SQL
Serial.println(F("Prueba de coexión con el servidor"));

WiFi.begin(SSID, PASS);
Serial.print("Conectando...");
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("Conexión OK!");
Serial.print("IP Local: ");
Serial.println(WiFi.localIP());
}
void loop() {

  //Valores finales de humedad
  HUM = analogRead(SenCap);
  ValHum = map(HUM, ValAgua, ValAire, 100, 0);

  //Valores finales de Iluminancia
  luminosidad = lightMeter.readLightLevel();

  //IMPRESIÓN DE DATOS
  if(ValHum >= 100)
  {
    humedad = 100;
    Serial.print("Humedad del suelo: ");Serial.print(humedad);Serial.println("
%");
  }
  else if(ValHum <=0)
  {
    humedad = 0;
    Serial.print("Humedad del suelo: ");Serial.print(humedad);Serial.println("
%");
  }
  else if(ValHum >0 && ValHum < 100)
  {
    humedad = ValHum;
    Serial.print("Humedad del suelo: ");Serial.print(humedad);Serial.println("
%");
  }
  Serial.print("Iluminancia: ");Serial.print(luminosidad);Serial.println(" lux");
  delay(2000);//pausa de 2 segundo solo para ver de mejor forma los valores
expuestos en el monitor serial
  //GENERACIÓN BASE DE DATOS_SQL
  if (WiFi.status() == WL_CONNECTED){
    HTTPClient http; // Objeto http
```

Código II.1 Código generado en el IDE de Arduino para realizar el envío de datos a MySQL utilizando el método HTTP POST (Parte 3)

```
String datos_a_enviar = "luminosidad=" + String(luminosidad) + "&humedad=" +
String(humedad); //Se envia Las variables

    http.begin(client,"http://mouldy-chalk.000webhostapp.com/EspPost.php");
//Inicializamos el protocolo HTTP y se indica la ubicación de la aplicación php
a la cual se envían los datos

//La ubicación se genera una vez creado el dominio del sitio web
    http.addHeader("Content-Type", "application/x-www-form-urlencoded"); //
Se define la estructura en la que se enviarán los datos

    int codigo_respuesta = http.POST(datos_a_enviar); // Se almacena la
respuesta del servidor en una variable entera, indicando si las solicitudes al
servidor han sido completadas

    if (codigo_respuesta>0){
        Serial.println("Código HTTP: " + String(codigo_respuesta));
        if (codigo_respuesta == 200){
            String cuerpo_respuesta = http.getString();//Se obtiene el mensaje
enviado por el servidor
            Serial.println("El servidor a respondido correctamente: ");
            Serial.println(cuerpo_respuesta);
        }
    } else {
        Serial.print("El servidor NO responde, código: ");
        Serial.println(codigo_respuesta);
    }

    http.end(); // Se libera recursos

} else {
    Serial.println("Conexion WIFI erronea");
}
delay(30000); //espera 30 segundos entre muestras
}
```


Código II.2 Aplicación php para conectarse con la base de datos

```
<?php

    $user = "id20151216_ax1";
    $pass = "Psicodelicial_";
    $server = "localhost";
    $db = "id20151216_bd_esp32";
    $con = mysqli_connect($server, $user, $pass, $db);

if ($con) {
    echo "Exito al conetarse con la base de datos! ";

    if(isset($_POST['luminosidad'])) {
        $luminosidad = $_POST['luminosidad'];
        // echo "Iluminancia del Jardin";
        echo " Iluminancia : ".$luminosidad;
    }

    if(isset($_POST['humedad'])) {
        $humedad = $_POST['humedad'];
        echo " humedad : ".$humedad;
        date_default_timezone_set('america/bogota');
        $fecha_actual = date("Y-m-d H:i:s");

        $consulta = "INSERT INTO BD_ESP32(Iluminancia, Humedad,
fecha_actual) VALUES ('$luminosidad','$humedad', '$fecha_actual')";
        // $consulta = "UPDATE DHT11 SET
Temperatura='$temperatura',Humedad='$humedad' WHERE Id = 1";
        $resultado = mysqli_query($con, $consulta);
        if ($resultado){
            echo " Registro en base de datos es exitosa! ";
        } else {
            echo "El registro en la base de datos a fallado";
        }
    }
} else {
    echo "ERROR! al conectarse con Base de datos ";
}

?>
```

ANEXO III. Manual de Conexión del ESP32 con la Plataforma de Arduino Cloud IoT

Se presenta una serie de pasos que permitirán replicar claramente la forma en que se conecta el ESP32 con la plataforma de Arduino IoT Cloud mediante Wi-Fi [57].

Paso 1: Ingresar a IoT Cloud

Ingresar a la página oficial de Arduino en <https://www.arduino.cc/>, para generar un perfil personal e ingresar al IoT Cloud, dando clic el icono que se muestra en la [Figura III.1](#).

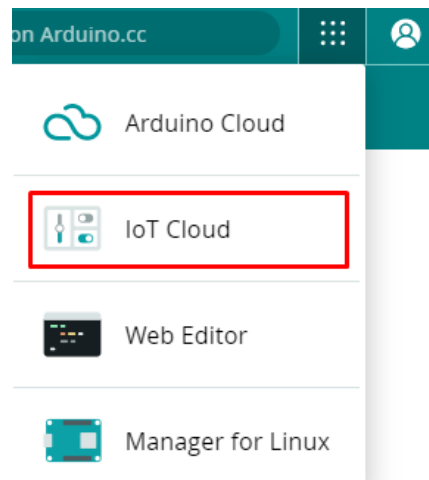


Figura III.1 Icono de IoT Cloud

Paso 2: Configuración de dispositivo

- 1) Ingresar en la pestaña “Dispositivos” para añadir un nuevo dispositivo como se ve en la [Figura III.2](#).

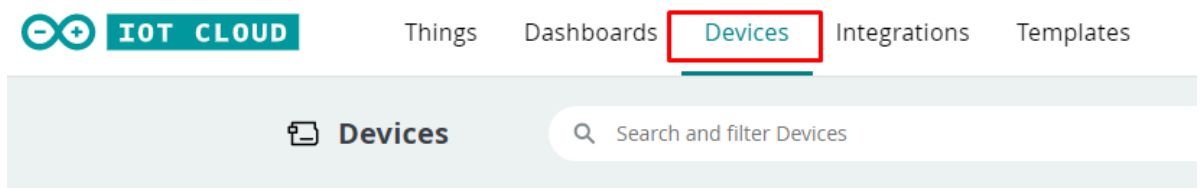


Figura III.2 Pestaña “Dispositivos”

- 2) Seleccionar el tipo de configuración según el dispositivo a ser utilizado, al utilizar el ESP32 se escoge “Configurar un dispositivo de terceros” como se ve en la [Figura III.3](#).

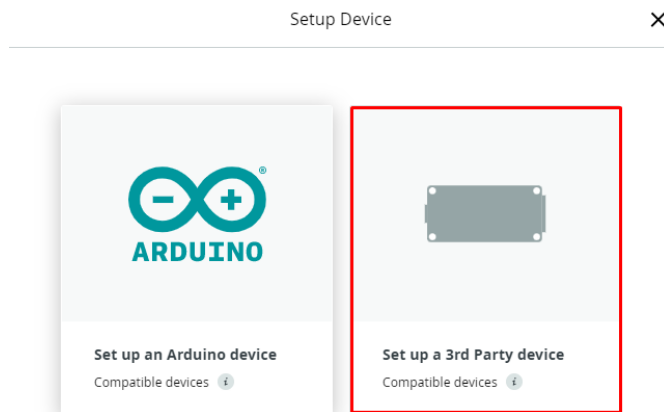


Figura III.3 Tipo de configuración seleccionado “Configurar un dispositivo de terceros”

- 3) Seleccionar el tipo de dispositivo y el modelo correspondiente, para el caso se utiliza el ESP32 modelo NodeMCU-32S como se ve en la [Figura III.4](#).

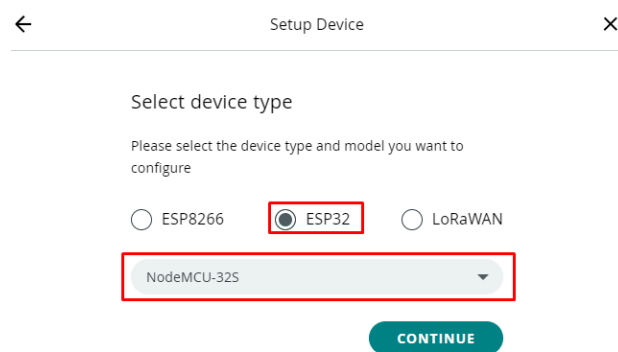


Figura III.4 Dispositivo ESP32 modelo NodeMCU-32S

- 4) Seleccionar el nombre del dispositivo, donde el nombre seleccionado para este dispositivo es “MaintESP32” como se muestra en la [Figura III.5](#).

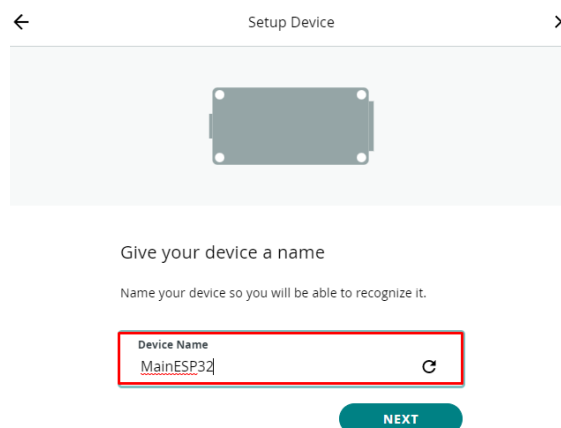


Figura III.5 Nombre del dispositivo “MaintESP32”

- 5) Se genera la identificación del dispositivo y su clave secreta como se ve en la [Figura III.6](#), estos datos son relevantes para la conexión el dispositivo con la nube y se debe reservarlos cuidadosamente dado que la clave secreta no podrá ser recuperada posteriormente.

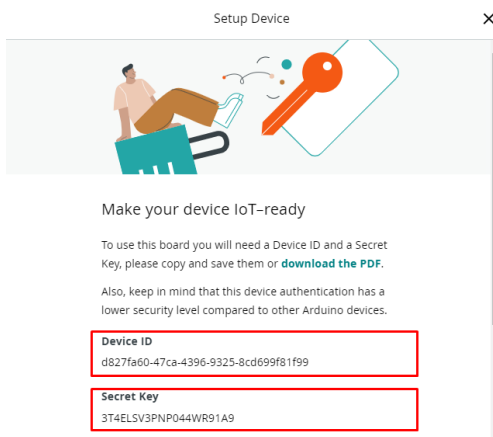


Figura III.6 Identificación del dispositivo y su clave secreta

Paso 3: Generar una “Cosa”

Ingresar en la pestaña “Things” para crear una cosa como se ve en la [Figura III.7](#).

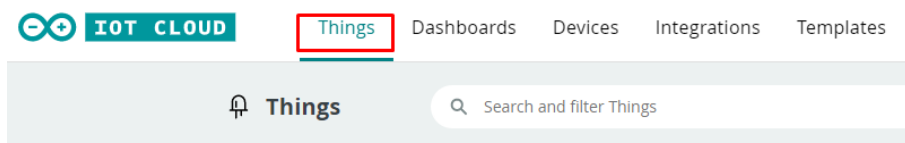


Figura III.7 Pestaña “Things”

Al añadir una nueva “Cosa” se tiene cuatro parámetros importantes a ser configurados, estos se muestran en la [Figura III.8](#).

- **Nombre:** Seleccionar un nombre adecuado para la “Cosa” según los parámetros de ingreso o salida que se tienen, para este caso se selecciona el nombre “Sensores”.
- **Dispositivo vinculado:** Se vincula el dispositivo “MainESP32” generado con antelación.
- **Variables:** Se crea las variables que tomarán los valores generados por los sensores, estas se llaman “humedad” y “luminosidad”, el tipo de datos de las dos variables es de tipo “int”, con permisos de “solo lectura” y la actualización de las variables está en “proceso de cambio”.

Red: Se ingresa el nombre de la red a la que se desea conectar j junto con su clave, además de la clave secreta generada al configurar el dispositivo.

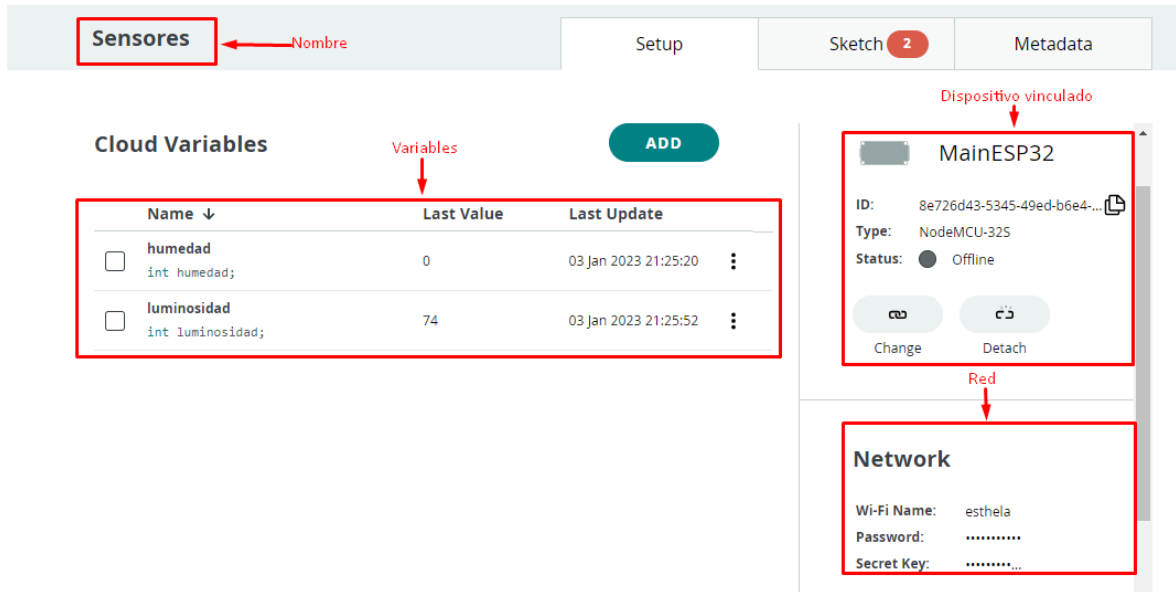


Figura III.8 Parámetros de configuración para la creación de una Cosa

Paso 4: Programado de la placa por el IDE de Arduino

Para programar la placa se da clic en la pestaña “Sketch” y luego a “Open full editor” como se ve en la [Figura III.9](#) dirigiéndose a el editor de Arduino en línea, donde se tiene ya una sección de código autogenerated por Arduino IoT Cloud el cual permite vincular las variables con el “dashboard” que será generado.

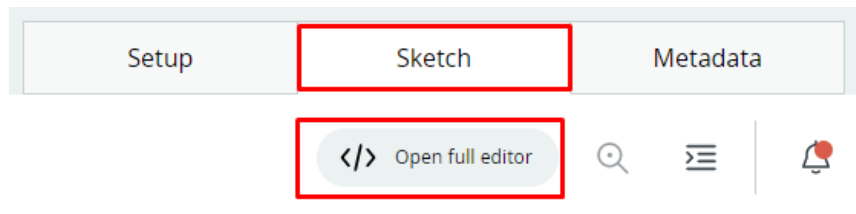


Figura III 9 Pestaña “Sketch”

El código autogenerated se encuentra en una librería llamada "thingProperties.h" como se ve en la [Figura III.10](#), este puede ser llamado desde el mismo “Sketch” del editor de Arduino en la web o en su defecto copiarlo y pasarlo al Arduino IDE instalado en el computador.

```

1 // Code generated by Arduino IoT Cloud, DO NOT EDIT.
2
3 #include <ArduinoIoTCloud.h>
4 #include <Arduino_ConnectionHandler.h>
5
6 const char DEVICE_LOGIN_NAME[] = "8e726d43-5345-49ed-b6e4-fb7f11eedfe5";
7
8 const char SSID[] = SECRET_SSID; // Nombre del inicio de sesión de del si
9 const char PASS[] = SECRET_OPTIONAL_PASS; // Contraseña de red (usar par
10 const char DEVICE_KEY[] = SECRET_DEVICE_KEY; // Contraseña secreta del dispositivo ES
11
12
13 void initProperties(){
14
15     ArduinoCloud.setBoardId(DEVICE_LOGIN_NAME);
16     ArduinoCloud.setSecretDeviceKey(DEVICE_KEY);
17     ArduinoCloud.addProperty(luminosidad, READ, ON_CHANGE, NULL);
18
19 }
20

```

Figura III.10 Sketch y librería “thingProperties.h” para conectar el ESP32 con Arduino IoT Cloud

Paso 5: Creación del “Dashboard”

Al ejecutar el *sketch* realizado se obtiene los valores de las variables vinculadas a los sensores y estos se los puede mostrar de una forma clara y didáctica en el *Dashboard*.

- 1) Para esto se dirige a la pestaña *Dashboards* como se ve en la [Figura III.11](#).

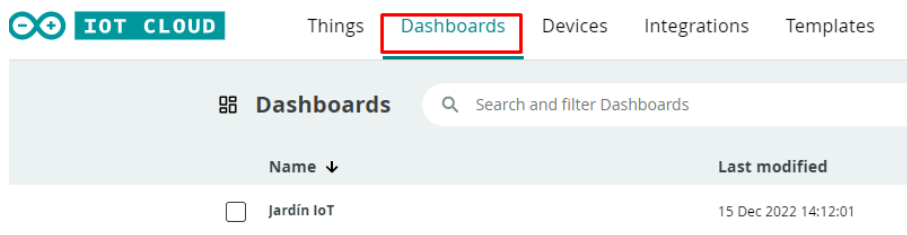


Figura III.11 Pestaña Dashboards

- 2) Seleccionar "Agregar *widjets*" ([Figura III.12](#)), en esta pestaña se encuentran varios componentes los cuales pueden ser vinculadas a las variables generadas y mostrar o mandar datos de una forma más simple e interactiva. Para el caso se ha seleccionado un “*Gauge*”, un “*Percentage*” y dos “*Chart*” los cuales permiten mostrar los valores generados por los sensores en tiempo real y monitorear sus cambios.

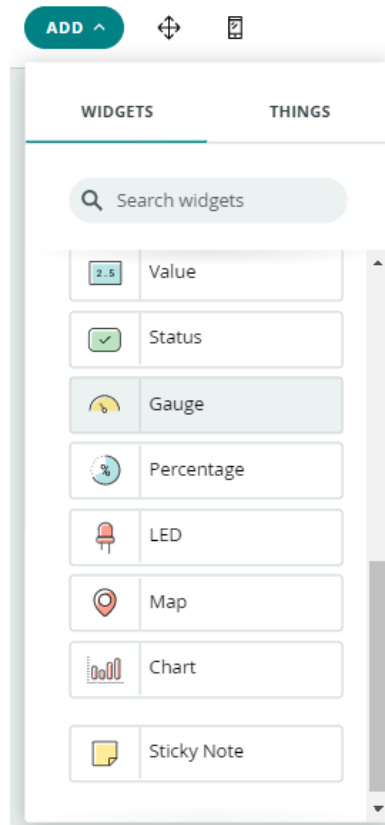


Figura III.12 Widgets disponibles

- 3) Se vincula las variables de humedad y luminosidad que fueron generadas previamente como se ve en la [Figura III.13](#).

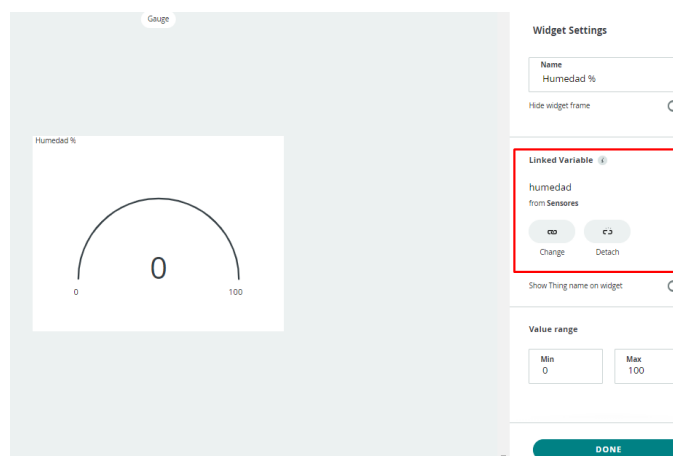


Figura III.13 Vinculación de Variables

- 4) Se coloca los widgets generados de la forma deseada, tanto para computadora como para celular como se ve en la [Figura III.14](#).

Jardín IoT

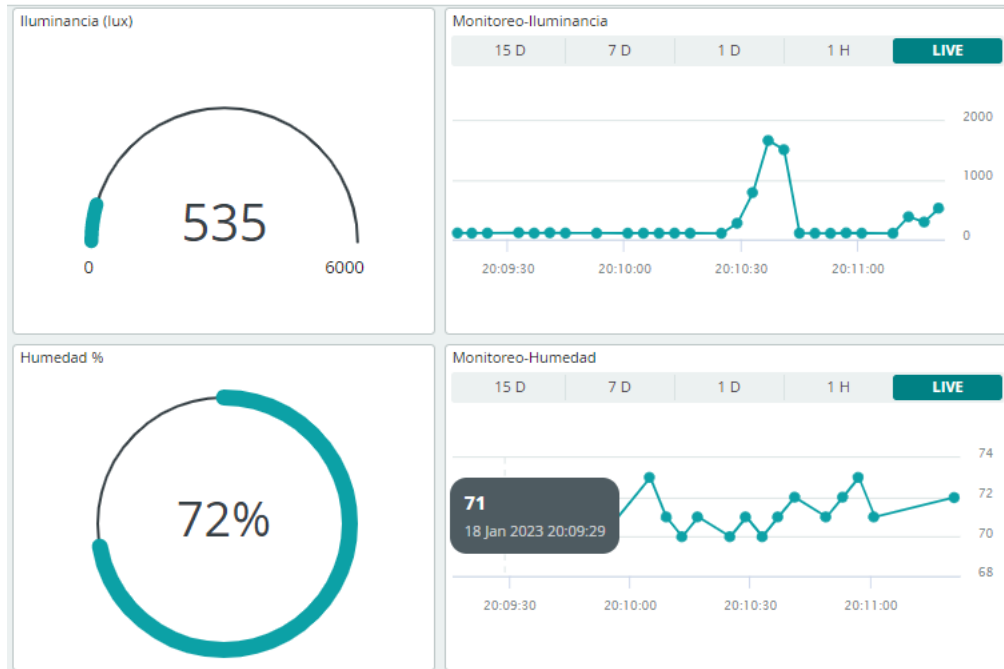


Figura III.14 Colocación de widgets

ANEXO IV. Envío de alerta por email a través de la plataforma IFTTT

Se presenta una serie de pasos que permitirán replicar la forma en la que se genera la rutina que permitirá el envío automático de correos de alerta a través de la plataforma IFTTT.

- 1) Se dirige hacia la página principal de IFTTT <https://ifttt.com/explore> y en el explorador se busca Webhooks ([Figura IV.1](#)), nos conectamos para luego dirigirse a la Documentación.

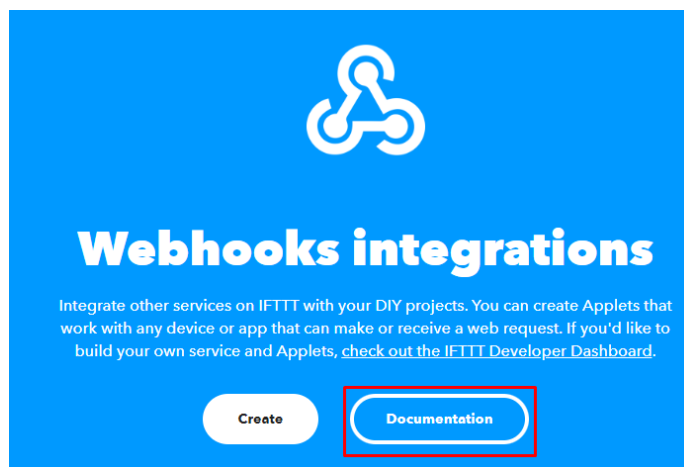


Figura IV.1 Webhooks

La documentación nos dirige hacia el link que desencadenara un evento con una carga útil JSON arbitraria. El evento generado se lo coloca en la parte de evento, que está señalado en la [Figura IV.2](#).



Figura IV.2 Documentación de Webhooks

- 2) Luego se ingresa a explorar y se busca Email, donde se conectará al correo electrónico al que se quiere enviar la alerta, o bien se conectará directamente al registrarnos previamente en la plataforma IFTTT. Para así dar clic en la opción crear, la cual nos permite generar una acción que nos permite enviar el correo como se ve en la [Figura IV.3](#).

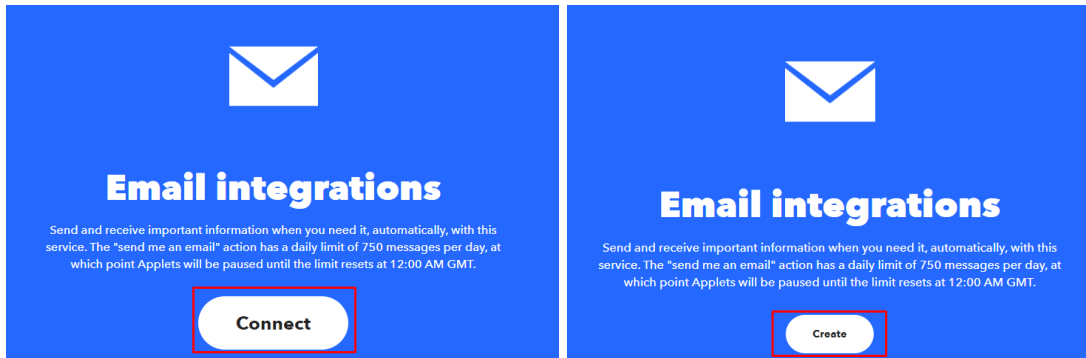


Figura IV.3 Email (Sección de conectar y crear)

Se nos despliega la ventana de “If This, Then That” mostrada en la [Figura IV.4](#).

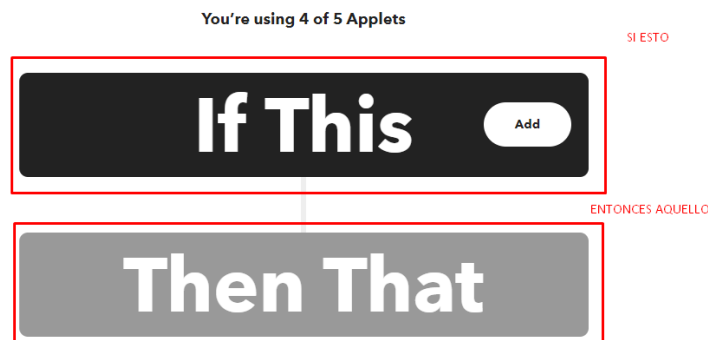


Figura IV.4 “If This, Then That”

- 1) Al dar clic “If This” y se selección Webhooks para realiza la petición, por lo que se da clic en la opción de recibir una petición web ([Figura IV.5](#)).

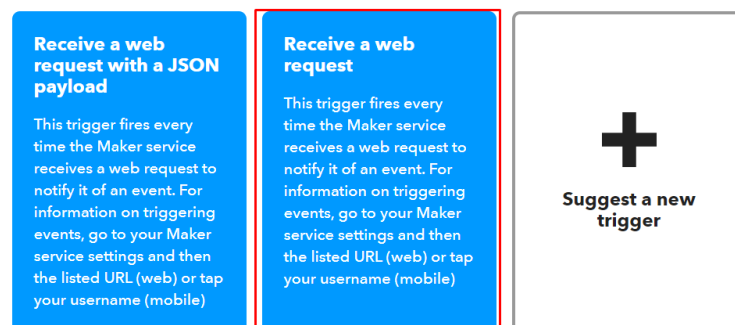


Figura IV.5 Recibir una petición web

Se le da un nombre al evento como se puede ver en la [Figura IV.6](#).

Figura IV.6 Nombre de la petición en “If This” (Sensor_iluminancia)

- 2) Se genera una acción en “Then That” la cual será enviar un correo, se selecciona email, aquí se especifica el asunto y el contenido del correo a ser enviado ([Figura IV.7](#)).

Figura IV.7 Acción generada por “Then That”, se especifica el contenido del correo

- 3) Finalizamos el proceso y ya tenemos los eventos “If This, Then That” establecidos. Se coloca el nombre de la petición de “If This” (Sensor_iluminancia) en la documentación de *Webhooks* en la parte del evento como se mostraba en la [Figura IV.2](#) y debe quedar como esta en la [Figura IV.8](#).

Realice una solicitud web POST o GET a:

```
https://maker.ifttt.com/trigger/Sensor_iluminancia / json / con/clave/j4ohQx8yhUJU5jJeldbPMeF1R_gaU3M-X0vs1mM3yeW
```

Figura IV.8 Solicitud web finalizada

ANEXO V. Enlace de Pruebas de Funcionamiento

Enlace: https://epnecuador-my.sharepoint.com/:v/g/personal/axel_cuamacas_epn_edu_ec/EfYi2FYtTSVOt-C2JCz9CWgBt7k2NdasDqUOdsXn-VW7Gw?e=BoEC0J



Figura V.1 Código QR de las pruebas de funcionamiento