



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**SISTEMA DE POSGRADO
MAESTRÍA EN TELECOMUNICACIONES**

TEMA

**Diseño de un sistema de monitorización de las variables
climáticas en huertos urbanos de la costa utilizando
plataforma abierta arduino y ubidots.**

AUTOR

Ing. Llivicura Gordillo, Fidel Antonio

**Trabajo de Titulación previo a la obtención del grado de
MAGISTER EN TELECOMUNICACIONES**

TUTOR

Ing. Bohórquez Heras, Daniel Bayardo, MSc

Guayaquil, Ecuador

11 de abril del 2025



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO
MAESTRÍA EN TELECOMUNICACIONES**

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por **Llivicura Gordillo, Fidel Antonio**, como requerimiento parcial para la obtención del Título de Magíster en Telecomunicaciones.

TUTOR

Ing. Bohórquez Heras, Daniel Bayardo, MSc

DIRECTOR DEL PROGRAMA

Ing. Bohórquez Escobar, Celso Bayardo, PhD

Guayaquil, 11 de abril del 2025



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO
MAESTRÍA EN TELECOMUNICACIONES**

DECLARACIÓN DE RESPONSABILIDAD

Yo, Llivicura Gordillo, Fidel Antonio

DECLARO QUE:

El trabajo de Titulación, **Diseño de un sistema de monitorización de las variables climáticas en huertos urbanos de la costa utilizando plataforma abierta arduino y ubidots**, previa a la obtención del Título de **Magíster en Telecomunicaciones**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, 11 de abril del 2025

AUTOR

f. 
Llivicura Gordillo, Fidel Antonio



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO
MAESTRÍA EN TELECOMUNICACIONES**

AUTORIZACIÓN

Yo, Llivicura Gordillo, Fidel Antonio

Autorizo a la Universidad Católica de Santiago de Guayaquil a la **publicación**, en la biblioteca de la institución del Trabajo de Titulación. **Diseño de un sistema de monitorización de las variables climáticas en huertos urbanos de la costa utilizando plataforma abierta arduino y ubidots**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, 11 de abril del 2025

AUTOR

f. 
Llivicura Gordillo, Fidel Antonio



UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO MAESTRÍA EN TELECOMUNICACIONES

REPORTE COMPILATIO



Reporte Compilatio del trabajo de titulación de la Maestría en Telecomunicaciones denominado: **"Diseño de un sistema de monitorización de las variables climáticas en huertos urbanos utilizando plataforma abierta Arduino y ubidots"**, del estudiante Ing. Fidel Antonio Llivicura Gonzalez se encuentra al 4% de coincidencias.

ING. DANIEL BOHÓRQUEZ HERAS, MGS.

DOCENTE-TUTOR

Agradecimientos

Al Director del Programa de Posgrado de la Universidad Católica Santiago de Guayaquil Ing. Bayardo Bohórquez Escobar, PhD, a mis profesores, a mi tutor, a mi familia y a mis amigos que contribuyeron para la creación de este proyecto de investigación.

Dedicatoria

A mis padres que ya partieron de este mundo a la presencia de Dios que son mi guía y dirección para alcanzar un peldaño más en mi formación académica y profesional. A mi esposa Mirian y a mis hijos Diego, Victoria y Gabriela que son mi fuente de inspiración y fortaleza que me apoyaron en todo momento a dar este paso importante de mi vida.



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

**SISTEMA DE POSGRADO
MAESTRÍA EN TELECOMUNICACIONES**

TRIBUNAL DE SUSTENTACIÓN

f. _____
Ing. Bohórquez Heras, Daniel Bayardo, MSc
TUTOR

f. _____
Ing. Bohórquez Escobar, Celso Bayardo, PhD
DIRECTOR DEL PROGRAMA

f. _____
MSc. Peñafiel Olivo, Kety Jenny
REVISOR

f. _____
MSc. Ubilla González, Ricardo Xavier
REVISOR

INDICE

INDICE.....	IX
RESUMEN	XIV
CAPITULO I: INTRODUCCIÓN	1
1.1. Antecedentes	1
1.1.1 Sugerencias de contenido	2
1.2. Justificación	3
1.3. Definición de problema.....	5
1.4. Planteamiento del problema.....	6
1.5. Objetivos.....	7
1.5.1. Objetivo General	7
1.5.2. Objetivos Específicos.....	7
1.6. Hipótesis	7
1.7. Metodología de investigación	8
CAPITULO II: MARCO TEORICO CONCEPTUAL	8
2.1 Agricultura Inteligente.....	8
2.2. Agricultura de precisión	9
2.3. Agricultura Urbana	11
2.4. La temperatura y las plantas	11
2.5. Sistemas de Supervisión Agrícola.....	12
2.6. Microcontroladores	13
2.7. Placa Arduino	14
2.8. Comunicación Inalámbrica	14
2.8.1. Clases de Redes Inalambricas WIFI	15
2.8.2. Ventajas y desventajas de una Red Inalambrica	15
2.8.3. Parametros de la red WIFI.....	16
2.9. Plataforma UBIDOTS	16
2.9.1. Beneficios de la Plataforma Ubidots	17
2.9.2. Creación de cuenta en Ubidots.....	17
2.9.6. Dashboard	20
2.9.7. Eventos.....	21
2.9.8. Configuración de Token.....	21

2.10. sensores y actuadores.....	22
2.10.1. Transductores.....	22
2.10.2. Clasificación de los sensores.....	22
2.10.3. Acondicionamiento de la Señal.....	23
2.11. Sensores utilizados en la Agricultura de Precisión	23
2.11.1. Sensor de Humedad del Suelo (Higrómetro)	23
2.11.2. Sensor de Temperatura y humedad	24
2.12. NodeMCU ESP8266	28
Arduino Mega	34
CAPITULO III: MARCO METODOLOGICO	35
3.1. Implementación del Hardware	35
3.1.1. Conexión del Higrómetro con tarjeta NodeMCU ESP32.....	35
3.1.2. Conexión del sensor de temperatura y humedad con la placa ESP32	36
3.1.3. Conexión del sensor de intensidad de luz (BH1750) con la placa ESP32	36
3.2. Desarrollo del código	36
3.4. Prueba en Sitio	49
3.4. Análisis de Costos	51
3.5. Discusión y análisis de los resultados	52
Conclusiones	53
Recomendaciones	54
REFERENCIAS BIBLIOGRÁFICAS.....	55

ÍNDICE DE TABLAS

Tabla 1 <i>Tipos de Placa Arduino</i>	14
Tabla 2 <i>Costos de los componentes</i>	51

ÍNDICE DE FIGURAS

Figura 1 Microcontrolador y sus partes	13
Figura 2 Página de acceso a Ubidots	18
Figura 3 Dispositivos de Ubidots	18
Figura 4 Variables de Ubidots	19
Figura 5 Datos del sensor de radiación solar	19
Figura 6 Datos del sensor DTH11	20
Figura 7 Dashboard de Ubidots	20
Figura 8 Creación de nuevo evento.....	21
Figura 9 Copia de Token predeterminado	21
Figura 10 Sensor HW-080 y Convertidor.....	24
Figura 11 Sensor de Temperatura y humedad.....	24
Figura 12 Sensor de Temperatura y humedad.....	25
Figura 13 Placa NodeMCU ESP32	27
Figura 14 NodeMCU ESP8266.....	29
Figura 15 Sensor Dióxido de Carbono MG811	31
Figura 16 Sensor de luz BH1750.....	32
Figura 17 Parte posterior del sensor BH1750	32
Figura 18 Entorno IDE Arduino	33
Figura 19 Diferentes tipos de Placa Arduino.....	33
Figura 20 Placa Arduino Mega	34
Figura 21 Conexión del Higrómetro con ESP32.....	35
Figura 22 Sensor DHT11 conectado con ESP32	36
Figura 23 Sensor BH1750 conectado con placa ESP32	36
Figura 24 Librerías utilizadas en el desarrollo del código	37
Figura 25 Declaración de funciones y configuración de red wifi	37
Figura 26 Almacenamiento de números de punto flotante y enteros utilizados en las variables climáticas.....	38
Figura 27 Desarrollo de código	39
Figura 28 Desarrolla el código	39
Figura 29 Desarrolla el código	40

Figura 30 Desarrolla el código	41
Figura 31 Desarrolla el código	41
Figura 32 Desarrolla el código	42
Figura 33 Desarrolla el código	42
Figura 34 Conexión de ESP 32 con los sensores de variables climáticas	49
Figura 35 Conexiones realizadas en protoboard y sensores ubicados en planta	50
Figura 36 Muestra del sensor de humedad del suelo en monitor serie del IDE de Arduino.....	50
Figura 37 Medición del Sensor de Humedad en Ubidots	51

RESUMEN

La investigación de este proyecto tiene como objetivos diseñar un sistema de bajo costo para monitorear las variables climáticas, desarrollar una interfaz intuitiva para visualizar los datos en tiempo real e implementar un sistema de alertas para prevenir problemas en los cultivos.

La metodología incluye la conexión de sensores de variables climáticas a un microcontrolador NodeMCUESP32 a través de sus pines Vcc, GND y OUT, los cuales transmiten datos en tiempo real a la plataforma Ubidots mediante MQTT, que es un protocolo de mensajería que se utiliza para conectar dispositivos y aplicaciones a través de internet. Finalmente, en Ubidots se creará un dashboard dinámico y widgets a través de la configuración que permite nombrar todos los dispositivos climáticos que se usarán para visualizar, monitorear y analizar la información recolectada, permitiendo una gestión eficiente del cultivo y la reducción de desperdicios.

Nuestro país se está abriendo paso hacia la tecnología 4.0 la misma que posee recursos como el internet de las cosas (IoT) , inteligencia artificial (IA), la automatización, el big data que son herramientas innovadoras para realizar procedimientos industriales en menos tiempo y con gran precisión.

Es el caso de la Empresa Claro que destina los recursos de la Tecnología 4.0 para ofrecer servicios de alimentación inteligente llamada BIOFEEDER que consiste en mejorar la calidad de la producción de camarones.

Otro ejemplo es la Empresa YARA que se encuentra establecida en nuestro país y ofrece agricultura de precisión orientado a las grandes empresas agrícolas.

La contribución de este estudio sobresale en la optimización del Diseño de un sistema que integra hardware, software, conectividad, IoT y configuración de plataforma en la nube que da lugar a una mejor producción agrícola.

Palabras claves: Cultivos, NodeMCUESP32, Arduino, Metodología, Ubidots, Automatización.

ABSTRACT

The research of this project aims to design a low-cost system to monitor climatic variables, develop an intuitive interface to visualize data in real time and implement an alert system to prevent crop problems.

The methodology includes the connection of climate variable sensors to an Arduino microcontroller through its Vcc, GND and OUT pins, which transmit data in real time to the Ubidots platform through MQTT, which is a messaging protocol used to connect devices. and applications over the internet. Finally, in Ubidots a dynamic dashboard and widgets will be created through the settings that allow you to name all the climatic devices that will be used to view, monitor and analyze the information collected, allowing efficient crop management and waste reduction.

Our country is making its way towards 4.0 technology, which has resources such as the Internet of Things (IoT), artificial intelligence (AI), automation, big data, which are innovative tools to carry out industrial procedures in less time and with great precision.

This is the case of the Claro Company, which allocates the resources of Technology 4.0 to offer intelligent feeding I or T services called BIOFEEDER, which consists of improving the quality of shrimp production.

Another example is the YARA Company, which is established in our country and offers precision agriculture aimed at large agricultural companies.

The contribution of this study stands out in the optimization of the Design of a system that integrates hardware, software, connectivity, IoT and cloud platform configuration that results in better agricultural production.

Keywords: Crops, NodeMCUESP32, Arduino, Methodology, Ubidots, Automation.

CAPITULO I: INTRODUCCIÓN

1.1. Antecedentes

El avance en el uso de los datos masivos en la agricultura ha aumentado progresivamente, aunque a un ritmo lento en el país. En contraste, en Estados Unidos, dicho avance ha sido más rápido, ya que algunas empresas (como Monsanto, DuPont, John Deere, entre otras), el sector gubernamental, privado y académico invierten recursos en el aprovechamiento de los datos que el sector genera y en el desarrollo de herramientas y soluciones para el campo.

Respecto a América Latina y el Caribe, se observa una evolución lenta que obedece a la baja implementación tecnológica. Lo anterior se evidencia con las cifras de la década de los 90, en las que se muestra una participación de aproximadamente el 8.3% en las exportaciones agroalimentarias mundiales, mientras que, en 2015, esta cifra se elevó solo al 13.8%, según la FAO.

Este escenario se presenta en un contexto de aumento poblacional global y obedece, en gran medida, a la evolución tecnológica de la Industria 4.0, también conocida como la cuarta revolución industrial. Este nuevo paradigma, que se refiere a la migración tecnológica y a las nuevas tendencias, ofrece un conjunto de aplicaciones que integran tecnologías como el Internet de las Cosas (IoT), la computación en la nube, los sistemas embebidos, los sensores inalámbricos, la inteligencia artificial y los sistemas de monitoreo y predicción.

Los huertos urbanos son necesarios en un país como Ecuador (y otros países) por diversas razones que abarcan aspectos ambientales, sociales, económicos y educativos, ya que contribuyen a la seguridad alimentaria, promueven la sostenibilidad ambiental, mejoran la calidad del aire y reducen el impacto del cambio climático, fomentan la educación y la concienciación ambiental, fortalecen el tejido social y comunitario e impulsan la economía local.

1.1.1 Sugerencias de contenido

Presentación del tema

La definición de huerto urbano según Ecólatras es un espacio abierto o cubierto en el que se pueden cultivar flores, plantas aromáticas, hortalizas, hierbas medicinales, frutales, a pequeña escala. (Ecológica, 2022)

Por lo tanto, se delimita el área de estudio con el propósito de diseñar un sistema de monitorización de las variables climáticas en cultivos de tomate en un huerto urbano de la Costa utilizando plataforma Arduino y ubidots.

“El tomate es originario de América del Sur y, aunque los aztecas ya lo consumían en Mesoamérica en el año 700 d.C., no fue hasta principios del siglo XIX cuando se convirtió en un popular ingrediente de la cocina mediterránea”. (Agricultura:, 2022)

Según Aprendamos una oportunidad para superarnos de la M.I. Municipalidad de Guayaquil:

Los tomates son cultivos frecuentes en el huerto casero. Incluyen la vitamina C y las del complejo B; son ricos en hierro y potasio. También contienen licopeno, que es un buen antioxidante y ayuda a prevenir los ataques al corazón, el cáncer de próstata y otros tipos de cáncer, según la clínica Mayo de los Estados Unidos. Además de tener un 90-95% de agua, el tomate se destaca por su contenido en beta-carotenos (compuestos que en el organismo se transforman en vitamina A). (2017, p.20)

La Industria 4.0 y su relación con la agricultura

La Industria 4.0 representa la cuarta revolución industrial, caracterizada por la integración de tecnologías digitales avanzadas en los procesos productivos y de gestión. Esta transformación implica la conexión de sistemas físicos y digitales a través de tecnologías como el Internet de las Cosas (IoT), la inteligencia artificial (IA), la robótica, el big data, la computación en la nube y la automatización. En el contexto de la agricultura, la Industria 4.0 ha dado lugar al concepto de Agricultura 4.0, que busca modernizar y optimizar los procesos agrícolas mediante el uso de estas tecnologías.

Beneficios de los huertos urbanos

Fomentar el ejercicio físico

Los huertos urbanos contribuyen al ejercicio físico. Porque una semilla al ser colocada en la tierra para que se convierta en fruto debe pasar por procesos de cavado, labrado, regado y cosechado, favoreciendo en la movilidad de las articulaciones, resistencia y fortaleza muscular en el ser humano y como resultado disminuye el sedentarismo y mejora la salud.

Mejora el bienestar emocional

El cuidado de las plantas nos permite entrar en contacto con la naturaleza y mejora la salud mental, disminuyendo el estrés, ansiedad y la falta de motivación.

Por lo tanto, trabajar en los huertos nos permite relajarnos, aumenta nuestro autoestima y salud mental.

Creación de espacios de armonía comunitaria

Promover la armonía comunitaria permite a las personas la colaboración, intercambiar ideas y robustecer las relaciones sociales.

Estos espacios fomentan el trabajo colaborativo, la incorporación y el sentido de pertenencia dentro de la comunidad, lo que implica mejorar el bienestar humano y lazo social.

1.2. Justificación

Enfoque de la problemática local

En Guayaquil, la expansión urbana y la falta de espacios verdes han limitado el desarrollo de huertos urbanos, lo que reduce oportunidades para el acceso a alimentos frescos, la educación ambiental y la integración comunitaria. Además, factores como el clima cálido, la contaminación y la escasez de agua pueden dificultar su sostenibilidad.

“Los Huertos caseros son: gran sostenibilidad ambiental debido a que la tierra enfrenta grandes retos para el uso eficiente de recursos para preservar y regenerar el capital natural, por eso es indispensable identificar espacios de inserción en las cadenas de valor, que permitan intercambios justos y equitativos gestionando responsablemente los recursos, estableciendo prácticas productivas de menor impacto, que permitan mitigar los impactos

del cambio climático” (Plan Nacional del buen vivir, 2017, p.81, citado por Helao & Morla, 2021)

Justificar la elección de la tecnología a usar

La elección de sensores remotos, sistemas integrados, computación en la nube y la plataforma Ubidots para la investigación sobre huertos urbanos en Guayaquil se justifica por los siguientes motivos:

Sensores remotos: Permiten monitorear variables clave como la humedad del suelo, la temperatura y la calidad del aire en tiempo real. Esto es crucial en un clima cálido como el de Guayaquil, donde el riego eficiente es esencial para la sostenibilidad de los cultivos.

Sistemas integrados: Facilitan la automatización del riego y otros procesos agrícolas, reduciendo el esfuerzo manual y optimizando el uso del agua. Esto es especialmente útil en zonas con acceso limitado a recursos hídricos.

Cloud computing: A través de la computación en la nube se puede guardar, examinar y visualizar datos desde cualquier parte del mundo, dando lugar al acceso continuo de la información y permitiendo la toma de decisiones de los datos en tiempo real.

Plataforma Ubidots: Ubidots está especializado en soluciones de hardware y software conectadas para monitorizar, controlar y automatizar procesos remotamente en el ámbito de la salud, energía, industria, fabricación, servicios públicos y transporte, con el fin de crear aplicaciones desde básicas hasta avanzadas. (Espinoza & Orellana, 2021)

Ubidots tiene un plan stem que es no comercial que facilita a los desarrolladores realizar entornos de computación en la nube que son factibles y fiables en un ambiente de plataforma I o T, aumentando la eficiencia y sostenibilidad de los huertos urbanos en Guayaquil.

Propuesta de mejora

En la ciudad existen pocos huertos urbanos y otros son auspiciados por La Municipalidad de Guayaquil y otros Organismos. Estos huertos se cultivan de manera tradicional con la siembra de la semilla y la persona o dueño debe seguir el control y cuidado durante el proceso de desarrollo de la planta o fruto.

En nuestro proyecto se propone con la agricultura inteligente realizar el proceso de monitoreo y control a través de un dispositivo móvil o computador para visualizar datos de las variables climáticas en tiempo real y determinar opciones para optimizar los cultivos.

1.3. Definición de problema

Guayaquil, ciudad ubicada en la región litoral de Ecuador, tiene un clima tropical monzónico, caracterizado por temperaturas cálidas durante todo el año y una temporada de lluvias bien definida entre diciembre y mayo. Esto la convierte en una región muy adecuada para huertos urbanos, donde se puede cultivar una amplia variedad de hortalizas, frutas, hierbas aromáticas y plantas ornamentales.

La nueva revolución agrícola es un concepto, cuyo significado y amplitud pueden provocar confusiones en su interpretación. Según Mooney (2020) los siguientes factores cubren todas las regiones y puntos de vista de la interpretación del concepto de la agricultura 4.0:

- La agricultura de precisión
- Industria 4.0
- Tecnología digital
- Toma de decisiones informada
- Barreras de la producción agrícola al aire libre
- Rentabilidad y sostenibilidad

Varias revisiones presentan una definición más amplia y clara de la idea de la agricultura 4.0, explicando que esta idea conlleva y desarrolla lo que antes se llamaba agricultura de precisión, que se lograba con la ayuda de sensores incluidos en la maquinaria utilizada dentro del área y otros sistemas, que se caracterizan por estar habilitados mediante el uso de tecnologías sabias y virtuales de la Industria 4.0. (Mooney, 2020, citado por Astudillo, 2022, p.7).

La mecanización agrícola tradicional, caracterizada por el uso de tractores y potencia motriz, se verá igualada e incluso superada por los equipos automatizados y la robótica y la precisión que pueden aportar en las operaciones agrícolas. (Agricultura, 2021)

Estas limitaciones son desafíos que se deben enfrentar, como los altos costos al implementar sistemas IoT, ya que se requieren redes estables (wi-fi, zigbee, LoRaWAN, Bluetooth, etc.). Además, el reemplazo de piezas y las actualizaciones de software incrementan los gastos.

Asimismo, la conectividad limitada en ciertas zonas urbanas, especialmente si no hay redes Wi-Fi confiables, y la interferencia en zonas densamente pobladas congestionan las redes.

Otra limitación es la falla en los dispositivos tecnológicos y la falta de conocimiento de los agricultores para manejar los cultivos manualmente.

1.4. Planteamiento del problema

La articulación del planteamiento del problema con la definición del problema nos garantiza coherencia y lógica en la argumentación detallada y se mencionan de la siguiente manera:

- Guayaquil tiene un clima tropical húmedo, caracterizado por altas temperaturas (20 a 30 °C) y una humedad elevada que al aplicar la tecnología 4.0 a través de sensores inteligentes nos permite monitorear estas variables climáticas.
- Las lluvias estacionales pueden provocar exceso de agua en los cultivos, mientras que las altas temperaturas aumentan la evaporación y el estrés hídrico en las plantas con el internet de las cosas se conectan dispositivos que regulan automáticamente el riego y la nutrición de las plantas según las condiciones climáticas.
- La contaminación urbana y la falta de suelo fértil agravan estos problemas y a través de Big Data y análisis predictivo se utilizan datos meteorológicos y del suelo para predecir patrones climáticos y ajustar estrategias de cultivo.
- La agricultura inteligente disminuye el trabajo manual y aumenta la productividad y eficiencia en los huertos urbanos..

1.5. Objetivos

1.5.1. Objetivo General

Diseñar un sistema de monitorización de las variables climáticas de los cultivos de tomate en los huertos urbanos utilizando plataforma Arduino y Ubidots.

1.5.2. Objetivos Específicos

1. Desarrollar un código eficiente en C++ eficiente para adquisición de datos de temperatura, humedad y radiación solar, utilizando sensores adecuados para las condiciones de los huertos urbanos en Guayaquil.
2. Calibrar los sensores utilizados para asegurar la precisión y confiabilidad de los datos recolectados en un rango de condiciones climáticas típicas de la región.
3. Diseñar una estructura de sistema que incorpora la placa ESP32, los sensores I o T y la plataforma en la nube, que permita una comunicación estable sin desbalance y en tiempo real.
4. Desarrollar una interfaz amigable e intuitiva en la plataforma Ubidots para visualizar y analizar los datos transmitidos desde los sensores de variables climáticas para crear aplicaciones de control y monitoreo.
5. Valorar la robustez del sistema en la toma de decisiones de los habitantes del sector agrícola y en la optimización de los productos y sostenibilidad de los huertos urbanos, a través de un estudio de caso o métodos de investigación.

1.6. Hipótesis

¿Cómo el diseño e implementación de un Sistema de monitoreo de variables climáticas en huertos urbanos es confiable en el sector agrícola?

Para que este sistema basado en agricultura inteligente sea infalible se debe proteger la seguridad y privacidad de los datos.

El agricultor debe tener acceso a la información que se van generando, y la transmisión de datos debe ser continua y sin restricciones.

1.7. Metodología de investigación

La presente investigación se enmarca en los siguientes enfoques metodológicos:

1. **Descriptivo:** Se basa en la descripción de los antecedentes del problema, utilizando fuentes bibliográficas como estudios previos, libros y otras investigaciones académicas que establecen un precedente para el análisis de proyectos afines.
2. **Cuantitativo:** Se enfoca en la medición, construcción, diseño y procesamiento de información, utilizando métodos numéricos y el método científico para analizar las variables.
3. **Cualitativo:** Se utiliza la comunidad de agricultores para entender el impacto de la agricultura urbana en la salud, la independencia y el bienestar espiritual. Esta metodología se realiza a través de entrevista y encuesta.

CAPITULO II: MARCO TEORICO CONCEPTUAL

Las variables climáticas son de gran utilidad para el normal desarrollo de los cultivos, por lo que la monitorización y control de estas deben ser consideradas para nuestro proyecto de agricultura inteligente.

2.1 Agricultura Inteligente

La agricultura inteligente es la adopción de tecnologías avanzadas y operaciones agrícolas basadas en datos para optimizar y mejorar la sostenibilidad de la producción agrícola. Las tecnologías utilizadas para la agricultura inteligente incluyen la inteligencia artificial (IA), la automatización y el Internet de las cosas (IoT). (Gomstyn & Jonker, 2023)

La preocupación por la seguridad alimentaria se ve agravada por el cambio climático, que afecta al rendimiento de los cultivos y pone en peligro la disponibilidad de recursos naturales como el agua para el riego. Además de lidiar con estos problemas, el sector agrícola también se enfrenta a desafíos de rentabilidad en medio del aumento de los costes de insumos como los

fertilizantes, la volatilidad de los precios de los productos básicos y el aumento de los requisitos reglamentarios. (Gomstyn & Jonker, 2023)

El fin de las nuevas agrotecnologías es, por tanto, llenar este vacío, generando información precisa capaz de influir en todos los resultados agrícolas. Son los datos, las mediciones y los conocimientos los que ayudan a elegir los mejores fertilizantes, las estrategias de rotación de cultivos y muchos otros aspectos de la gestión del campo. (agforest, 2023)

2.2. Agricultura de precisión

La agricultura de precisión (AP) comprende un sin número de técnicas relacionadas con el uso óptimo de los insumos agrícolas en función de la variabilidad espacial y temporal de la producción. La AP es definida como la habilidad para monitorear y controlar con precisión las actividades dentro de una empresa agrícola, y también poseer los conocimientos suficientes para comprender todos los procesos relacionados, de modo que puedan aplicarse los resultados obtenidos, para llegar a una meta. (Pereira, 2014)

Busca aumentar la productividad de las actividades agrícolas a nivel mundial, en este marco de desarrollo la navegación autónoma es un pilar fundamental ya que la mayoría de actividades agrícolas implican desplazamientos extensos de personas o vehículos; la navegación autónoma terrestre sobre parcelas de cultivos tiene ciertas complicaciones como el deslizamiento de los vehículos sobre los terrenos lo que complica implementar sistemas de edometría, otra complicación es la dificultad de implementar marcas de navegación para usar sistemas de localización. (Moreano, 2020, citado por Sánchez, Martínez, & Torres, 2024, p.6)

La agricultura de precisión analiza los factores climáticos para manejar un buen procedimiento. Entre los factores climáticos sobresale:

Temperatura: Es un factor importante en el cultivo y crecimiento de las plantas para la producción de las cosechas. Junto con los niveles de luz, dióxido de carbono, humedad del aire, agua y nutrientes. La estabilidad de estos factores es primordial. La temperatura afecta a la planta tanto a corto como a largo plazo. (CANNA, s.f., citado por Jiménez, 2020, p.6)

Viento: Los efectos causados por este factor son clasificados como directos o indirectos. Los directos incluyen el movimiento de la planta, el daño físico de hojas y frutos, aborto de flores, la rotura de ramas, el vuelco o descalzado cuando la fuerza ejercida por el viento excede la resistencia del tallo o de la raíz. Los indirectos son aquellos que son transportados por el viento u otros factores como la arena, (Alicia Graciela Kin & Jean François Ledent, s. f., citado por Jiménez, 2020, p.7)

Radiación Solar: El sol es una fuente primaria de energía radiante, caracterizada por transmitir ondas a través del espacio. Por su parte, las plantas teniendo la capacidad de sintetizar todas las sustancias esenciales para su metabolismo a partir de sustancias inorgánicas, actúan como excelentes captadores de la radiación solar. Por ello, mediante el proceso de la fotosíntesis transforman la energía calorífica en energía química. Así, quedan asociadas a los compuestos orgánicos de la biomasa. (Sistema Agrícola, 2016, citado por Jiménez, 2020, p. 7)

Precipitación: Las precipitaciones afectan la humedad del suelo donde crecen y se desarrollan los cultivos y es importante registrar las muestras de las mismas para realizar planificaciones efectivas. Estas planificaciones van, desde programar una siembra hasta la cosecha del cultivo. (agrotecnología, s.f.,citado por Jimenez, 2020, p.7)

Humedad: El déficit de humedad del suelo disminuye la disponibilidad de los nutrimentos a pesar de que se encuentren en cantidades suficientes. Las plantas requieren que los nutrimentos se encuentren disueltos en la solución del suelo para que estos puedan ser absorbidos y translocados hasta los lugares donde van a ser metabolizados. El exceso de agua o el déficit en la zona radicular afectan la forma química en la que están presentes los nutrientes en el suelo. (Muñoz, 2009)

2.3. Agricultura Urbana

El Programa de Naciones Unidas para el Desarrollo define la agricultura urbana como la actividad que produce, procesa y comercializa alimentos y otros productos, en suelo y en agua, en áreas urbanas y periurbanas, aplicando métodos de producción intensivos y utilizando y reciclando recursos naturales y de desecho, para producir diversidad de cultivos y ganado. (Naciones Unidas, 1996, citado por Hernández, 2006, p. 16)

2.4. La temperatura y las plantas

Una variable fundamental en los cultivos es la temperatura que es fundamental para el desarrollo de las plantas, también con los niveles de luz, dióxido de carbono, humedad del aire, agua y nutrientes, la temperatura influye en el crecimiento de la planta y la productividad de las cosechas. Todos estos factores deberían estar equilibrados. La temperatura afecta a la planta tanto a corto como a largo plazo. (Canna, 2021)

La mayoría de los procesos biológicos se acelerarán con temperaturas altas, lo cual puede ser tanto positivo como negativo. Un rápido crecimiento o producción de frutos es un beneficio en la mayoría de los casos, sin embargo, la excesiva respiración que se produce es desfavorable porque implica que quedará menos energía disponible para el desarrollo de los frutos, resultando en unos frutos más pequeños.

Las modificaciones de las propiedades térmicas del suelo, con el mulch, plantas de cobertura, la labranza, los camellones y el riego acercan la temperatura al rango óptimo de la planta. De los factores agrometeorológicos, la radiación solar tiene el efecto más grande sobre el calor edáfico y es modificado por la latitud, altitud, exposición del terreno al sol, y tipo y color del suelo. Conforme aumenta la altitud disminuye la temperatura del aire, siendo la temperatura del suelo más alta que la ambiental, a partir de los 500 msnm hacia arriba. La temperatura edáfica está calificada como crucial para el crecimiento y la adaptación de plantas a la altitud. (Fischer & Torres, 1999)

2.5. Sistemas de Supervisión Agrícola

Son sistemas tecnológicos que recogen información de los cultivos para colaborar en la decisión de la gestión de la granja. Estos sistemas emplean sensores, satélites, GPS y software para administrar el estado de la siembra. Algunos ejemplos de sistemas de monitoreo agrícola son:

ASIS (Agriculture Stress Index System)

El problema de la sequía es muy importante para la agricultura debido a su impacto en la seguridad alimentaria mundial, la Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO) ha desarrollado una herramienta para monitorear y detectar a nivel global las áreas agrícolas donde los cultivos podrían ser afectados por sequía. Esta herramienta se denomina ASIS o Sistema del Índice de Estrés Agrícola, el cual se basa en el Índice de Salud Vegetal (VHI) y es generado a partir de información satelital. El sistema es operacional en FAO-Roma a nivel global como apoyo técnico del Sistema Global de Información y Alerta Temprana (Global Information and Early Warning System, GIEWS, por sus siglas en inglés). (Ordoñez, Rojas, & Hernandez, 2019)

GPS (Global Positioning System)

El sistema de posicionamiento global (GPS), es un instrumento de navegación compuesta por 24 satélites a 20 000 kilómetros por encima de la tierra con orbitas móviles, enviando datos precisos de la hora y posición de satélites, lo que permite calcular la localización exacta del receptor en tierra. (Antichan y Ñoñez 2019, citado por León, 2022, p.1)

El uso de esta herramienta (GPS), trae consigo beneficios a los agricultores, siendo su principal función el facilitar la jornada laboral del mismo, entre sus utilidades la principal es la orientación de tractores, ya que la utilización de estos sistemas favorece a la conducción de vehículos agrícolas, minimizando los solapamientos que se producen al ejecutar las diversas labores agrícolas sin sistemas de orientación o enrutamiento (Pérez 2012, citado por León, 2022, p.1).

BushelFarm

Es una aplicación móvil que permite a los agricultores y ganaderos llevar un registro de sus actividades diarias, monitorear el clima, gestionar inventarios

y programar tareas. Facilita la toma de decisiones basada en datos y mejora la productividad en la gestión de explotaciones ganaderas. (Bushelfarm, 2022, citado por Ortega, 2023, p.7).

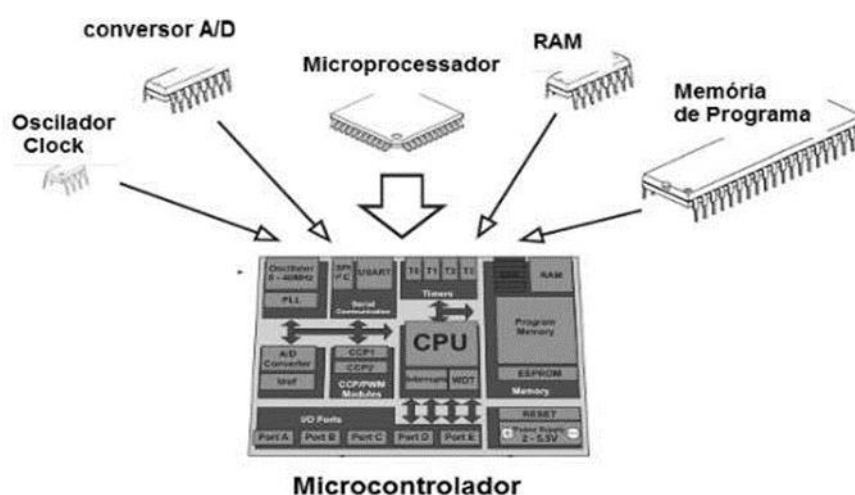
Bushel Farm fue creada en 2011 como FarmLogs, y en 2021 fue adquirida por Bushel, una plataforma de comercio de granos.

2.6. Microcontroladores

Un microcontrolador es un circuito integrado digital monolítico que contiene todos los elementos de un procesador digital secuencial síncrono programable de arquitectura Harvard o Princeton (Von Neumann). Se le suele denominar también microcomputador integrado o empotrado (Embedded processor) y esta especialmente orientado a tareas de control y comunicaciones. (Mandado, Menéndez, Fernández, & López, 2007).

Por su pequeño tamaño, los microcontroladores permiten empotrar un procesador programable en muchos productos industriales. Su coste reducido y su consumo de energía y velocidad adaptables, resultan apropiados para numerosas aplicaciones. Además, poseen mecanismos de seguridad de funcionamiento (safety) y proporcionan protección del equipo electrónico contra copias y modificaciones del programa no autorizadas (security). (Mandado, et al., 2007).

Figura 1 Microcontrolador y sus partes



Nota: Adaptado Microcontrolador, por Reinoso y Bohórquez., 2024.

2.7. Placa Arduino

Arduino es un controlador que se basa en software y hardware fáciles de usar de código abierto (open source). El hecho de ser una plataforma de código abierto le hace destacar frente a otras, ya que los planos de los módulos están publicados bajo licencia. Además, ofrece la facilidad de ejecutarse en sistemas operativos varios, como son Windows, Linux o Macintosh OSX. (Vila, 2018)

Arduino ofrece una variedad de modelos de placa, a continuación, las mas usadas se presenta en la tabla:

Tabla 1 *Tipos de Placa Arduino*

MODELO	CARACTERISTICA
ARDUINO UNO	Es el primer modelo, la mayoría de placas restantes posee las características de este modelo. Consta de 14 pines digitales, 6 pueden usarse como PWM (Modulación por ancho de pulso) y hasta 6 pines analógicos.
ARDUINO MEGA	Es un modelo potente de Arduino. Respecto a las propiedades eléctricas es muy similar a la placa Arduino uno aunque posee 54 pines digitales, 15 de ellos PWM y 16 pines analógicos.
ARDUINO NANO	Aunque su tamaño es reducido, se trata de un modelo muy completo, con conexión MINI-USB pero sin conector de alimentación externa, Su creación fue dirigido a aplicaciones de bajo coste.
ARDUINO PRO	Este modelo fue creado para usuarios que buscan flexibilidad y precios bajos.

Fuente: Vila (2018)

Elaborado por: El Autor

2.8. Comunicación Inalámbrica

La comunicación Wi-Fi (Wireless) es un sistema que permite crear redes inalámbricas. Esta tecnología esta direccionada a la interconectividad de equipos informáticos como PCs, organizadores (PDA), consola de video juegos e incluso las impresoras.

2.8.1. Clases de Redes Inalambricas WIFI

- **Ad-Hoc:** Este tipo de red carece de dispositivos de infraestructura como servidores o enrutadores, es decir los dispositivos inalámbricos se conectan entre sí y comparten recursos e incluso internet.
- **Infraestructura:** Este tipo de red posee puntos de acceso o dispositivos de infraestructura como servidores, conmutadores o enrutadores que les permite gestionar las redes de manera óptima y segura.

2.8.2. Ventajas y desventajas de una Red Inalambrica

La principal ventaja en una red inalámbrica es la movilidad y la ubicuidad, por lo que se elimina toda la infraestructura de cables en la comunicación. La flexibilidad es otra ventaja, así como la accesibilidad ya que la gran parte de dispositivos informáticos tienen acceso a la tecnología wifi y pueden ser parte de alguna red inalámbrica y compartir recursos de manera segura desde donde quiera que se encuentren.

La movilidad es una ventaja que permite a las personas conectarse a la red desde cualquier punto de la región sin necesidad de estar en sus puestos de trabajo, haciéndolos más productivos. El aumento de productividad implica aumento en la información y en las aplicaciones de la Organización. Los clientes también pueden acceder sin riesgo a la plataforma de la Empresa para gestionar sus procesos.

Es asequible la configuración de las redes inalámbricas, ya que su instalación es sin demora y rentable porque no hace uso de cableado físico. Este tipo de redes tienen la capacidad de adaptabilidad y escalabilidad lo que le permite reducir costes sin perder eficiencia a la hora de hacer cambios.

Ahora, una de las desventajas es la interferencia. Esto se produce por dispositivos electrónicos informáticos que están dentro o fuera de la red inalámbrica y operan con frecuencias similares. Por ejemplo, los inhibidores de señal no permiten una buena comunicación o a veces se pierde la comunicación.

Otra desventaja es la seguridad de la información, ya que se puede acceder a información protegida o no protegida que hace vulnerable a una red y en consecuencia afecta a una Empresa o Corporación.

2.8.3. Parametros de la red WIFI

- **SSID.** – Acrónimo de Service Set Identifier (Identificador de conjunto de servicios), es decir se identifica a una red wifi con un nombre para diferenciarlas de las otras redes.

WEP. - Acrónimo de Wired Equivalent Privacy (Privacidad equivalente por cable). Es un protocolo de seguridad inalámbrico, tuvo su origen en 1999, se utilizaba para cifrar datos y preservar la red wifi pero no era tan segura. En 2003 es lanzado WPA (Wi-Fi Protected Access) que también es un protocolo de seguridad pero con mejoras que no permite el acceso a terceros.

- **MAC.-** Es el acrónimo de Media Access Control (Control de Acceso al Medio). La dirección MAC está formada por seis bloques, los primeros tres bloques identifican al fabricante y los tres restantes al producto. Todos los dispositivos que intervienen en una red van a tener dirección MAC. Es decir sirve para gestionar el tráfico de entrada y salida de cada dispositivo identificado que se conecta con algún router o servidor. La combinación de estos tres parámetros, mejora la productividad, disminuye los costos operativos y aumenta el bienestar del usuario.

2.9. Plataforma UBIDOTS

Ubidots es una plataforma o servidor en la nube que toma los datos de los sensores I o T y los transmite a través de MQTT que es el protocolo de conexión para subirlo a la nube de ubidots.

Ubidots facilita a los desarrolladores de software crear aplicaciones de monitoreo y control dentro de la red IoT.

Debido al acceso inalámbrico, en la plataforma se presentan problemas de seguridad, siendo el más importante, la autenticación de usuario y el cifrado de la información.

Otro tipo de seguridad es el firmware que es un software informático que establece lógica de bajo nivel de los dispositivos. El HTTP con cifrado SSL (capa de sockets seguro) y el MQTT con cifrado TLS (transporte de mensajería ligero).

2.9.1. Beneficios de la Plataforma Ubidots

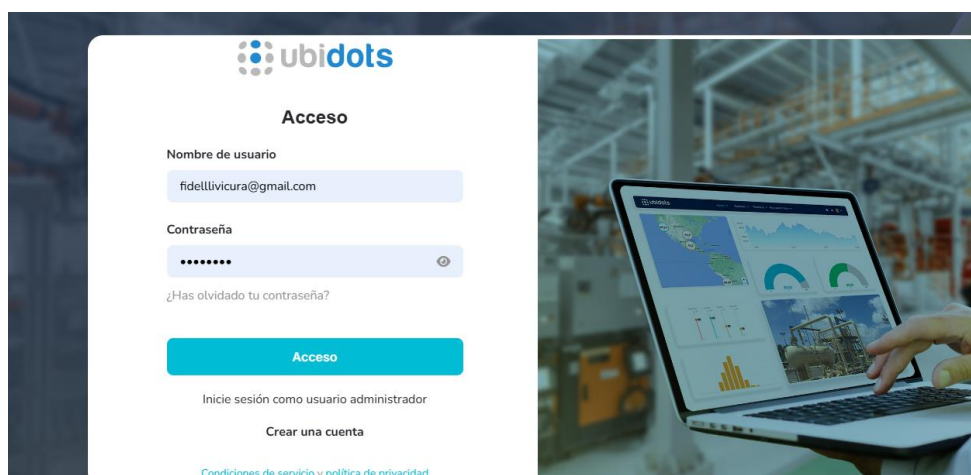
La plataforma Ubidots presenta los siguientes beneficios:

1. **Comparte librerías:** El uso de las librerías para Visual studio code o Arduino IDE que permite enviar o recibir datos de la plataforma sin ser un desarrollador experimentado.
2. **Seguridad:** Establece controlar a quienes ingresen a los datos de los dispositivos, otorga permisos de aplicaciones, organizaciones y usuarios y revoca el acceso de usuarios cuando sea necesario.
3. **Escalabilidad:** Escala desde un dispositivo hasta miles de dispositivos. Además escala soluciones en la nube desde proyectos piloto hasta proyectos llave en mano.
4. **Eficiencia:** Conecta fácilmente el hardware a Ubidots a través de las librerías que se encuentran en repositorios github y se utilizan protocolos de conexión como MQTT, HTTP, TCP, UDP.
5. **Integración:** Los elementos esenciales de cualquier aplicación I o T en Ubidots son: Dispositivos, Variables, Variables sintéticas, dashboards y eventos. Estos elementos al integrarse permiten agrupar, examinar y visualizar los datos.

2.9.2. Creación de cuenta en Ubidots

Para utilizar la nube se requiere ser cliente y se debe crear un usuario en su plataforma Ubidots STEM que proporciona licencia gratuita con un correo electrónico y un password, tal como se muestra en la figura 2. Luego de creado el usuario, estamos listos para trabajar en nuestros proyectos.

Figura 2 *Página de acceso a Ubidots*



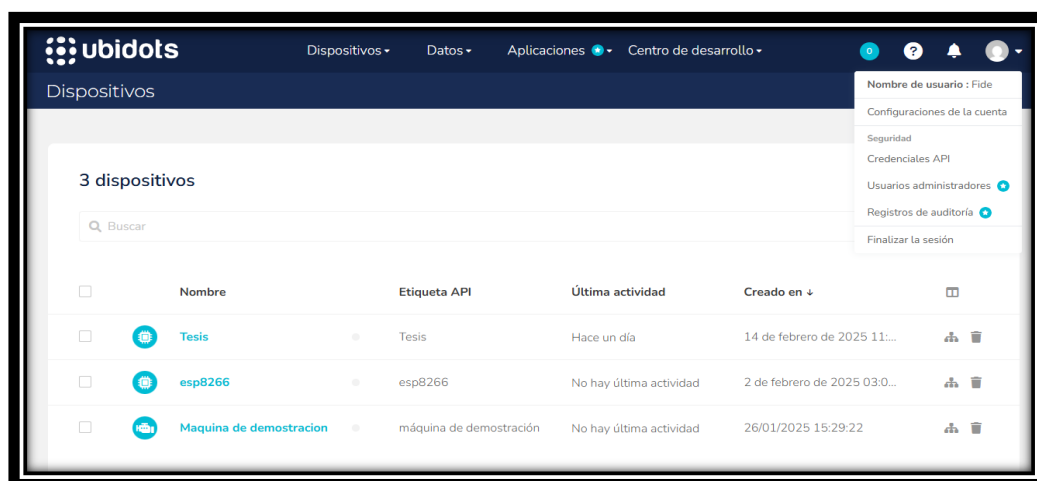
Fuente: www.ubidots.com

Elaborado: Por el Autor

2.9.3. Dispositivos

Para nuestro proyecto se crea el dispositivo con el nombre de Tesis el mismo que es una representación virtual y recoge los datos de tres sensores que continuamente están transmitiendo y se conecta a la plataforma de ubidots.

Figura 3 *Dispositivos de Ubidots*



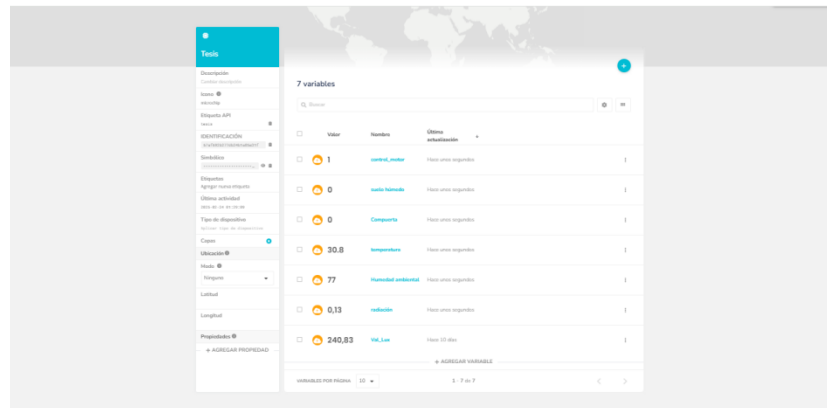
Fuente: www.ubidots.com

Elaborado: Por el Autor

2.9.4. Variables

Para este proyecto se crearon siete variables que almacenan de manera temporal datos, que cada cierto periodo de tiempo es actualizado. Estas variables son: humedad ambiental, temperatura, control de motor, suelo húmedo, radiación, val_lux, compuerta.

Figura 4 Variables de Ubidots



Fuente: www.ubidots.com

Elaborado: Por el Autor

2.9.5. Variables Sintéticas

Esta herramienta es el resultado de crear información a partir de los datos brutos almacenados en ubidots, los mismos que son generados con fecha, día y hora y optimiza la toma de decisión del usuario.

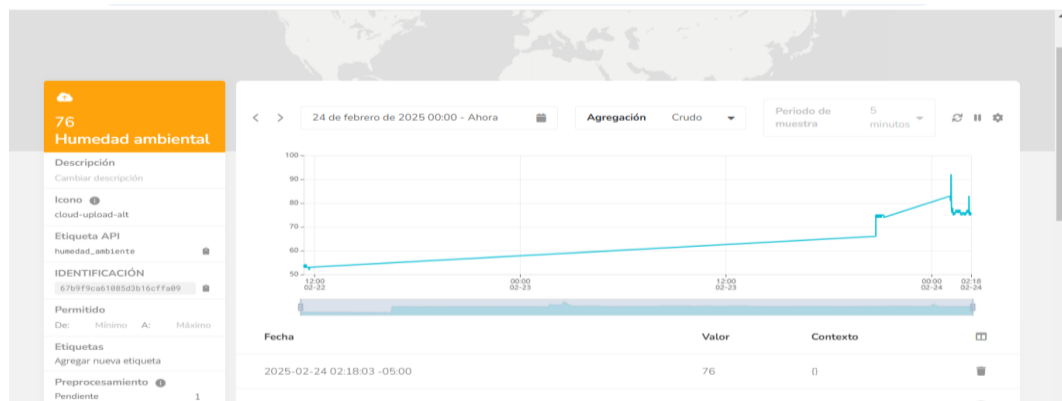
Figura 5 Datos del sensor de radiación solar



Fuente: www.ubidots.com

Elaborado: Por el Autor

Figura 6 Datos del sensor DTH11



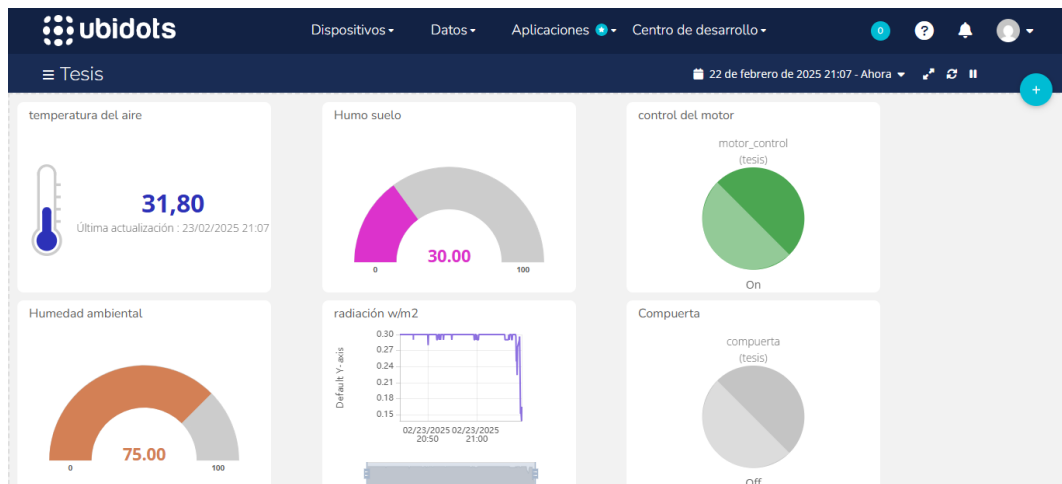
Fuente: www.ubidots.com

Elaborado: Por el Autor

2.9.6. Dashboard

Es el interfaz que permite observar los datos en tiempo real de manera ordenada y fácil. En este panel se visualiza la temperatura del aire, la humedad del suelo, la humedad ambiental, la radiación solar, el control de motor de la bomba de agua y la compuerta que se abre o se cierra si la radiación solar aumenta fuera del rango de operación.

Figura 7 Dashboard de Ubidots



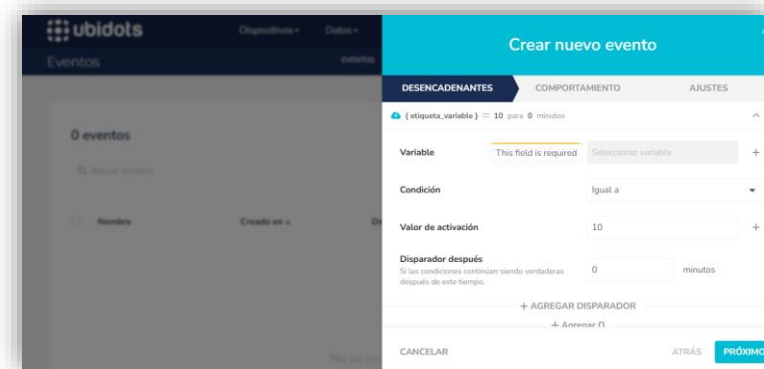
Fuente: www.ubidots.com

Elaborado: Por el Autor

2.9.7. Eventos

Los eventos son mensajes que se activan cuando se cumplen ciertas condiciones. Y estos mensajes son enviados a través de correo electrónico y mensaje de texto sms, para nuestro caso.

Figura 8 Creación de nuevo evento



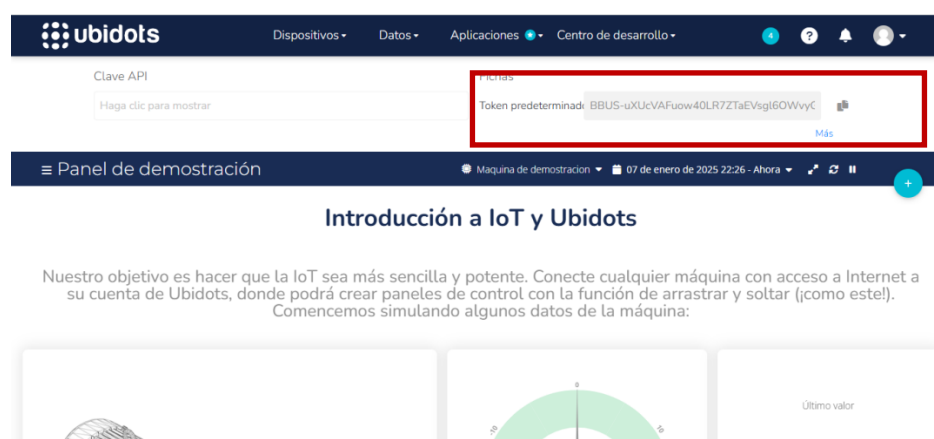
Fuente: www.ubidots.com

Elaborado: Por el Autor

2.9.8. Configuración de Token

El token es la única clave que a través de este los dispositivos pueden interactuar con el interfaz de programación de aplicaciones (API). Para poder realizar su configuración se debe desplegar en la parte superior derecha y seleccionar credenciales API, una vez aquí va a copiar en token predeterminado para pegar en el IDE de Arduino o en Visual studio code

Figura 9 Copia de Token predeterminado



Fuente: www.ubidots.com

Elaborado: Por el Autor

2.10. SENSORES Y ACTUADORES

2.10.1. Transductores

Un Transductor se define como aquel dispositivo que transforma el efecto de una causa física en otro tipo de señal que puede ser eléctrica. De acuerdo con esta definición, es posible afirmar que un transductor forma parte de un sensor o de un actuador, pero la diferencia entre un sensor, un actuador y un transductor radica en que el transductor simplemente cambia el dominio de la variable, mientras que el sensor proporciona una salida útil para ser usada como variable de entrada a un sistema de procesamiento de la información y el actuador se encarga de ejecutar la acción determinada por el sistema de procesamiento de la información. (Corona, Abarca, & Mares, 2014)

2.10.2. Clasificación de los sensores

Los sensores se pueden clasificar de varias formas, entre ellas, por el tipo de variable física que miden, por su señal de salida, o por si son activos o pasivos.

Por el tipo de variable física

Sensores mecánicos, Sensores químicos, Sensores eléctricos, Sensores térmicos, Sensores magnéticos.

Por su señal de salida

- Sensores analógicos: la salida varía de forma continua
- Sensores digitales: la salida varía en pasos discretos

Por si son activos o pasivos

Los sensores activos tienen algunos beneficios sobre los sensores pasivos, como poder operar en condiciones de poca luz u oscuridad, tener una mayor resolución y precisión, y poder medir la distancia, la velocidad y la dirección del objetivo. Sin embargo, los sensores activos también tienen algunos inconvenientes, como requerir más energía y generar más ruido, verse afectados por interferencias y desorden, y ser más detectables y vulnerables a las contramedidas. (Pani, 2025)

Los sensores pasivos tienen algunas ventajas sobre los sensores activos, como ser más sigilosos y encubiertos, consumir menos energía y generar menos ruido, y verse menos afectados por la interferencia y el desorden. Sin embargo, los sensores pasivos también tienen algunas limitaciones, como depender de la disponibilidad y calidad de la radiación o emisión natural, tener una resolución y

precisión más bajas, y no poder medir la distancia, la velocidad y la dirección del objetivo. (Pani, 2025)

2.10.3. Acondicionamiento de la Señal

El Acondicionamiento de señal es uno de los componentes fundamentales de un dispositivo moderno de adquisición de datos (también conocido como sistema DAQ o DAS). El propósito básico de un sistema de adquisición de datos es realizar mediciones físicas. Se componen de los siguientes componentes básicos: (Smith, 2024)

- Sensores
- Acondicionamiento de señal
- Convertidor analógico a digital (ADC)
- Y algún tipo de computadora con software DAQ para el registro y análisis de señales.

2.11. Sensores utilizados en la Agricultura de Precisión

Los sensores I o T son de gran importancia para el agricultor pues le permite monitorizar uno de los factores más importantes en la producción de su cultivo y controlar uno de los costes más importantes: El agua de riego.

Se ha demostrado que medir correctamente los valores de humedad a varias profundidades permiten reducir hasta un 30% de media el consumo de agua de riego a la vez que se maximiza la producción y su calidad. (PRISMAB)

Los sensores en la agricultura de precisión son dispositivos que miden la humedad, la temperatura, la salinidad, el pH y otros parámetros del suelo. Estos sensores permiten a los agricultores tomar decisiones más informadas para mejorar la productividad de sus cultivos.

A continuación, se presenta los principales sensores utilizados para optimizar la producción agrícola:

2.11.1. Sensor de Humedad del Suelo (Higrómetro)

Para nuestro proyecto sobre el control y monitorización de huertos urbanos es importante el uso de este sensor ya que permite evaluar la humedad del suelo y ayuda a programar el suministro y la distribución del agua de una manera óptima.

Este dispositivo consta de electrodos y del convertidor. Es decir, los electrodos actúan como resistencia, mientras más húmedo es el suelo menor es la resistencia. El convertidor AD se encarga de convertir los valores analógicos del sensor en datos digitales.

Figura 10 Sensor HW-080 y Convertidor



Nota. Adaptado Sensor de humedad de suelo (Fotografía), por AbrahamC., 2020.

2.11.2. Sensor de Temperatura y humedad

Sensor DHT22

Este dispositivo sirve para medir la temperatura y humedad del ambiente.

Es decir, recoge valores analógicos y a través del pin de datos sale un dato digital.

Figura 11 Sensor de Temperatura y humedad



Fuente: Naylamp Mechatronics

Elaborado por: El Autor

Especificaciones Técnicas DHT22

- Voltaje de Operación: 3V - 6V DC
- Rango de medición de temperatura: -40°C a 80 °C
- Precisión de medición de temperatura: $\leq \pm 0.5$ °C
- Resolución Temperatura: 0.1°C
- Rango de medición de humedad: De 0 a 100% RH
- Precisión de medición de humedad: 2% RH
- Resolución Humedad: 0.1%RH
- Tiempo de sensado: 2s
- Interface digital: Single-bus (bidireccional)
- Modelo: AM2302
- Dimensiones: 20*15*8 mm
- Peso: 3 gr.
- Carcasa de plástico blanco

PINES

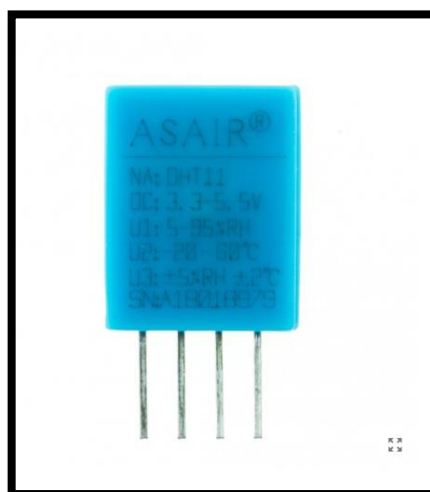
- 1- Alimentación: +5V (VCC)
- 2- Datos (DATA)
- 3- No Usado (NC)
- 4- Tierra (GND)

Sensor DHT11

Este dispositivo sirve para medir la temperatura y humedad del ambiente.

Es decir, recoge valores analógicos y a través del pin de datos sale un dato digital, el mismo que ingresa a la placa ESP32 de nuestro proyecto.

Figura 12 *Sensor de Temperatura y humedad*



Fuente: Naylamp Mechatronics

Elaborado por: El Autor

Especificaciones Técnicas DHT11

- Voltaje de Operación: 3V - 5V DC
- Rango de medición de temperatura: 0 a 50 °C
- Precisión de medición de temperatura: ± 2.0 °C
- Resolución Temperatura: 0.1°C
- Rango de medición de humedad: 20% a 90% RH.
- Precisión de medición de humedad: 5% RH.
- Resolución Humedad: 1% RH
- Tiempo de sensado: 1 seg.
- Interface digital: Single-bus (bidireccional)
- Modelo: DHT11
- Dimensiones: 16*12*5 mm
- Peso: 1 gr.
- Carcasa de plástico celeste

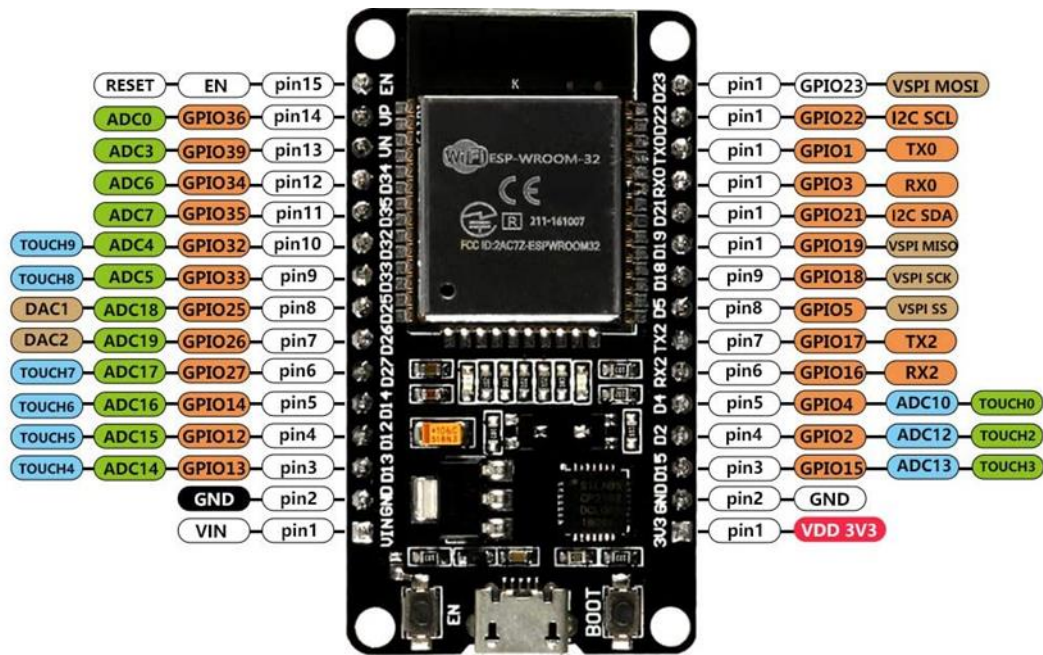
PINES

- 1- Alimentación: +5V (VCC)
- 2- Datos (DATA)
- 3- No Usado (NC)
- 4- Tierra (GND)

NodeMCU ESP32

Es una placa de desarrollo la cual contiene el chip System On Chip (SoC) cuyo nombre es ESP32 y fue desarrollado por Espressif System.

Figura 13 Placa NodeMCU ESP32



Fuente: Naylamp Mechatronics

Elaborado por: El Autor

Especificaciones Técnicas NodeMCU ESP32

- Voltaje de Alimentación (micro-USB): 5V DC
- Voltaje de Alimentación (Vin): 4V-12V DC (5V-9V recomendado)
- Voltaje de Entradas/Salidas: 3.3V DC
- Placa: ESP32 DEVKIT V1 (Espressif)
- SoM: ESP-WROOM-32 (Espressif)
- SoC: ESP32 (ESP32-D0WDQ6)
- CPU: Dual-Core Tensilica Xtensa LX6 (32 bit)
- Frecuencia de Reloj: hasta 240Mhz
- Desempeño: Hasta 600 DMIPS
- Procesador secundario: Permite hacer operaciones básica en modo de ultra bajo consumo
- Wifi: 802.11 b/g/n/e/i (802.11n @ 2.4 GHz hasta 150 Mbit/s)
- Bluetooth: v4.2 BR/EDR, Bluetooth Low Energy (BLE)
- Memoria: 448 KB ROM, 520 KB SRAM, 16 KB SRAM in RTC, 4 MB QSPI Flash/SRAM
- Pines: 30
- Pines Digitales GPIO: 24 (Algunos pines solo como entrada)

- Pines PWM: 16
- Pines Analógicos ADC: 18 (3.3V, 12bit: 4095, tipo SAR, ganancia programable)
- Conversor Digital a Analógico DAC: 2 (8bit)
- UART: 2
- Chip USB-Serial: CP2102
- Conector USB: micro-USB
- Antena en PCB
- Seguridad: Estándares IEEE 802.11 incluyendo WFA, WPA/WPA2 and WAPI, 1024-bit OTP, up to 768-bit for customers, Aceleración criptográfica por hardware: AES, HASH (SHA-2), RSA, ECC, RNG
- Dimensiones: 55*28 mm
- Peso: 9 gramos

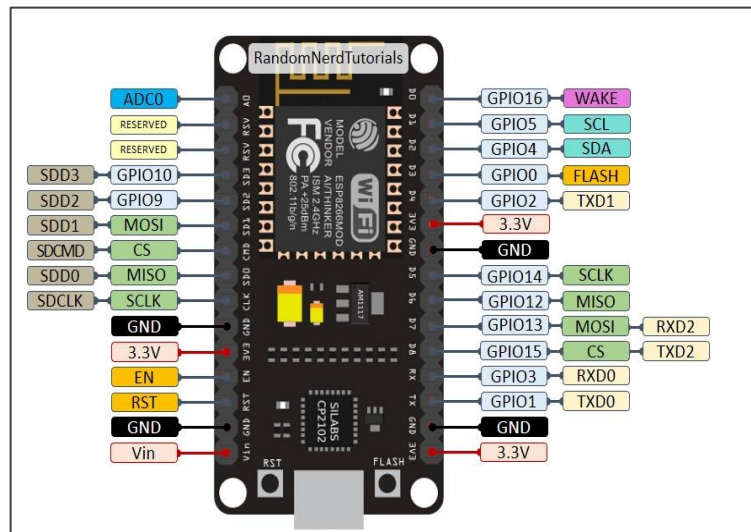
FUNCIONALIDADES

- LED POWER en placa para indicar el encendido
- LED LED_BUILTIN: GPIO2 (D2)
- Regulador de voltaje (3.3V) en placa
- Conector micro-USB para programación y comunicación serial integrado en placa
- Pines con espaciado de 0.1" para insertarlo directamente sobre un protoboard
- Pulsador ENABLE (RESET)
- Pulsador BOOT para seleccionar modo de inicio (Bootloader o Flash)
- RESET automático al subir un sketch
- El Bootloader integrado permite programarlo con el entorno de desarrollo Arduino sin necesidad de un programador externo

2.12. NodeMCU ESP8266

NodeMCU ESP8266 es una placa desarrolladora que sirve para implementar una plataforma de IoT (Internet of Things), de open source. Usa un firmware ejecutándose sobre el SoC ESP8266 de Espressif Systems.

Figura 14 NodeMCU ESP8266



Fuente: Exploradores

Elaborado por: El Autor

Especificaciones Técnicas

- Voltaje de Alimentación: 5V DC
- Voltaje de Entradas/Salidas: 3.3V DC (No usar 5V)
- Placa: NodeMCU v2 (Amica)
- Chip conversor USB-serial: CP2102
- SoM: ESP-12E (Ai-Thinker)
- SoC: ESP8266 (Espressif)
- CPU: Tensilica Xtensa LX3 (32 bit)
- Frecuencia de Reloj: 80MHz/160MHz
- Instruction RAM: 32KB
- Data RAM: 96KB
- Memoria Flash Externa: 4MB
- Pines Digitales GPIO: 17 (4 pueden configurarse como PWM a 3.3V)
- Pin Analógico ADC: 1 (0-1V)
- Puerto Serial UART: 2
- Certificación FCC
- Antena en PCB
- 802.11 b/g/n
- Wi-Fi Direct (P2P), soft-AP

- Stack de Protocolo TCP/IP integrado
- PLLs, reguladores, DCXO y manejo de poder integrados
- Potencia de salida de +19.5dBm en modo 802.11b
- Corriente de fuga menor a 10uA
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4ms guard interval
- Wake up and transmit packets in < 2ms
- Consumo de potencia Standby < 1.0mW (DTIM3)
- Pulsador RESET y FLASH
- Led indicadores: 2
- Dimensiones: 49*26*12 mm
- Peso: 9 gramos

Conectividad

- SDIO 2.0, SPI, UART
- Integra RF switch, balun, 24dBm PA, DCXO y PMU
- Posee un procesador RISC, memoria en chip e interface para memoria externa
- Procesador MAC/Baseband integrado
- Interface I2S para aplicaciones de audio de alta calidad
- Reguladores de voltaje lineales "low-dropout" en chip
- Arquitectura propietaria de generación de clock "spurious free"
- Módulos WEP, TKIP, AES y WAPI integrados

Sensor CO2 (MG811)

El sensor está diseñado para detectar dióxido de carbono en el medio ambiente. «El dióxido de carbono es un gas incoloro e inodoro que es esencial para la vida en la Tierra. El compuesto natural consta de dos enlaces covalentes dobles en un átomo de carbono y dos átomos de oxígeno. El CO2 existe como un gas menor en la atmósfera terrestre en una concentración de aproximadamente 0,04% (400 ppm) por volumen. Las fuentes naturales incluyen volcanes, fuentes termales, géiseres, que se liberan cuando las rocas carbonatadas se disuelven en agua y ácidos.

Debido a que el dióxido de carbono es soluble en agua, se encuentra naturalmente en aguas subterráneas, ríos, lagos, casquetes polares, glaciares y océanos. Se encuentra en campos de petróleo y gas. (Novatronic)

Figura 15 *Sensor Dióxido de Carbono MG811*



Fuente: Novatronic

Elaborado por: El Autor

Descripción de sensor de Intensidad de luz (BH1750)

EL módulo BH1750 es un sensor de luz, que a diferencia del LDR es digital y nos entrega valores de medición en Lux (lumen /m^2) que es una unidad de medida estándar para el nivel de iluminación (iluminancia). Tiene alta precisión y un rango ente 1 – 65535 lx el cual es configurable.

EL módulo tiene un regulador interno de 3.3V pudiendo alimentar con 5V sin problemas.

La interfaz de comunicación es I2C pudiéndolo implementar en la mayoría de microcontroladores, el módulo aparte de los pines de alimentación y pines I2C tiene un pin para establecer la dirección.

Figura 16 *Sensor de luz BH1750*



Fuente: Naylamp Mechatronic

Elaborado por: El Autor

Figura 17 *Parte posterior del sensor BH1750*



Fuente: Naylamp Mechatronic

Elaborado por: El Autor

2.10. Programación en entorno IDE Arduino

Arduino se programará mediante un editor de texto o "sketch" el cual usa un lenguaje propio creado expresamente para la plataforma Arduino, este lenguaje, está basado en "Processing", que es un lenguaje de alto nivel. El Sketch es un entorno de programación sencillo el cual no tiene más que las opciones básicas de programación, verificación del código y carga en la placa Arduino. Véase en la figura 9 la programación IDE.

Figura 18 Entorno IDE Arduino



Fuente: Wikipedia

Elaborado por: El Autor

Placa Arduino

Arduino forma parte de los sistemas embebidos. Esta placa esta compuesta por microcontrolador programable y circuitos electrónicos. Es una plataforma open source que se utiliza para realizar proyectos prototipo y proyectos llave en mano.

Figura 19 Diferentes tipos de Placa Arduino

Modelo	Microcontrolador	Voltaje de entrada	Voltaje del sistema	Frecuencia de Reloj	Digital I/O	Entradas Analógicas	PWM	UART	Memoria Flash	Interfaz de Programación
Arduino Due	AT91SAM3X8E	5-12V	3.3V	84MHz	54	12	12	4	512Kb	Nativa USB
Arduino Leonardo	ATmega32U4	7-12V	5V	16MHz	20	12	7	1	32Kb	Nativa USB
Arduino Uno - R3	ATmega328	7-12V	5V	16MHz	14	6	6	1	32Kb	USB via ATmega16U2
Arduino Pro 3.3V/8MHz	ATmega328	3.3V-12V	3.3V	8MHz	14	6	6	1	32Kb	Cabecera compatible con FTDI
Arduino Pro 5V/16MHz	ATmega328	5-12V	5V	16MHz	14	6	6	1	32Kb	Cabecera compatible con FTDI
Ethernet	ATmega328	7-12V	5V	16MHz	14	6	6	1	32Kb	Cabecera compatible con FTDI
Arduino Mega 2560 R3	ATmega2560	7-12V	5V	16MHz	54	16	14	4	256Kb	USB via ATmega16U2
Arduino Mini 05	ATmega328	7-9V	5V	16MHz	14	6	8	1	32Kb	Cabecera Serial
Arduino Pro Mini 3.3V/8MHz	ATmega328	3.3V-12V	3.3V	8MHz	14	6	6	1	32Kb	Cabecera compatible con FTDI
Arduino Pro Mini 5V/16MHz	ATmega328	5-12V	5V	16MHz	14	6	6	1	32Kb	Cabecera compatible con FTDI
Arduino Flo	ATmega328P	3.3V-12V	3.3V	8MHz	14	8	6	1	32Kb	Cabecera compatible con FTDI o Inalámbrica via XBee ¹

Fuente: Aprendiendo Arduino

Elaborado por: El Autor

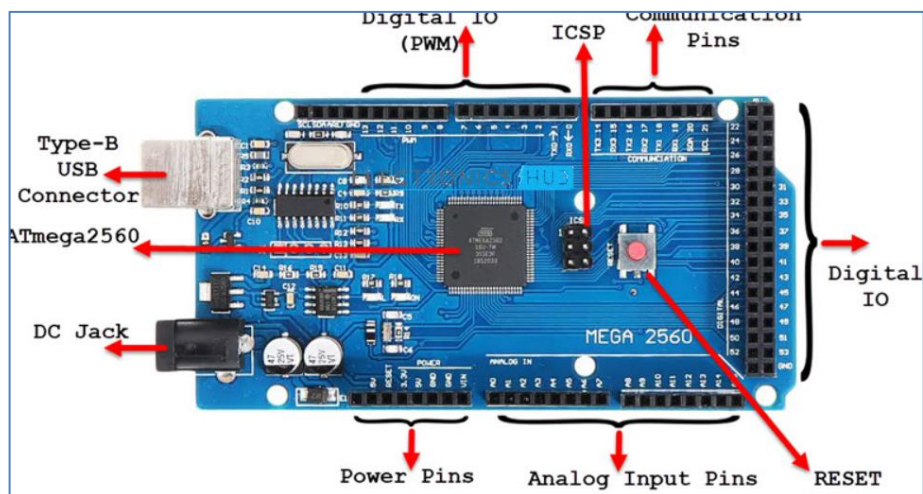
Arduino Mega

Las principales características de un Arduino son:

- ✓ Microcontrolador: ATmega328
- ✓ Voltaje Operativo: 5v
- ✓ Voltaje de Entrada (Recomendado): 7 – 12 v
- ✓ Pines de Entradas/Salidas Digital: 14 (De las cuales 6 son salidas PWM)
- ✓ Pines de Entradas Análogas: 6
- ✓ Memoria Flash: 32 KB (ATmega328) de los cuales 0,5 KB es usado por Bootloader.
- ✓ SRAM: 2 KB (ATmega328)
- ✓ EEPROM: 1 KB (ATmega328)
- ✓ Velocidad del Reloj: 16 MHZ.

A continuación, se muestra las partes de un Arduino Mega en la figura 19.

Figura 20 Placa Arduino Mega



Fuente: Arduino Fórum

Elaborado por: El Autor

CAPITULO III: MARCO METODOLOGICO

3.1. Implementación del Hardware

En las siguientes subsecciones se procede con el detalle de las conexiones entre tarjeta NodeMCU ESP32 y los sensores.

3.1.1. Conexión del Higrómetro con tarjeta NodeMCU ESP32

A continuación, se presentan las conexiones de la tarjeta ESP32 con el higrómetro, son cuatro:

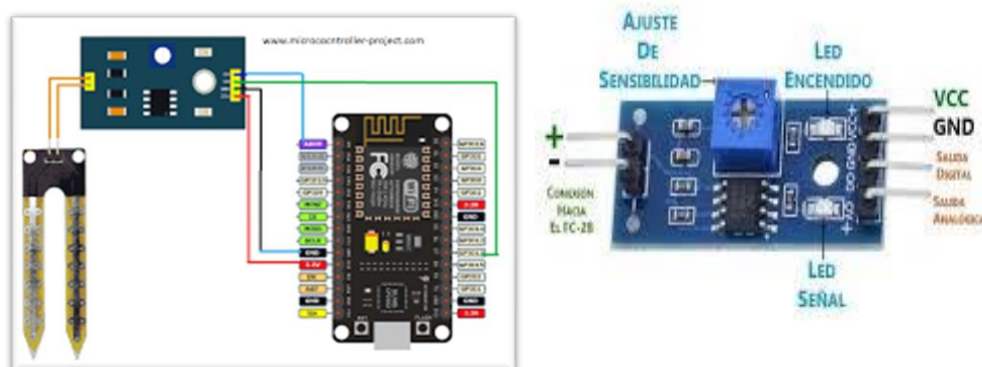
Vcc, voltaje de alimentación (3,3 V – 5 VDC)

GND Tierra

D0 Salida Digital

A0 Salida Analógica

Figura 21 *Conexión del Higrómetro con ESP32*



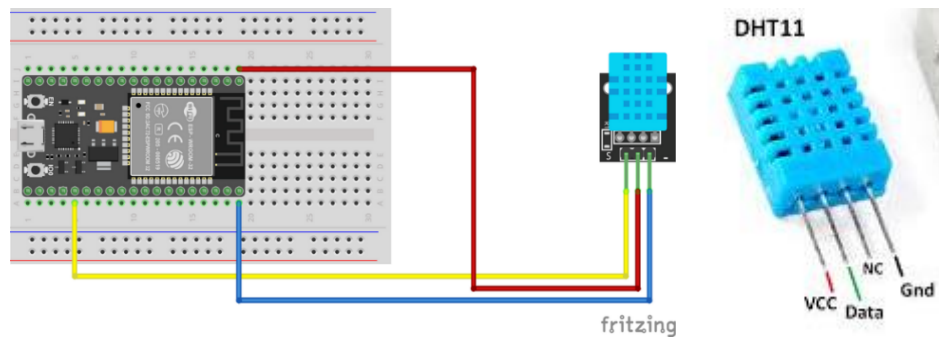
Fuente: Compraco

Elaborado por: El Autor

Se configura el sensor de humedad con un rango de 1023 para 100 divisiones y el nivel de humedad a un valor 60 que es en donde se activa un ventilador o una bomba de agua, por lo tanto, la bomba de agua se activará con un 40% de humedad o menos en caso de que se conecte un relé activador del sistema de bombeo y recibe la señal por el pin digital como se muestra.

3.1.2. Conexión del sensor de temperatura y humedad con la placa ESP32

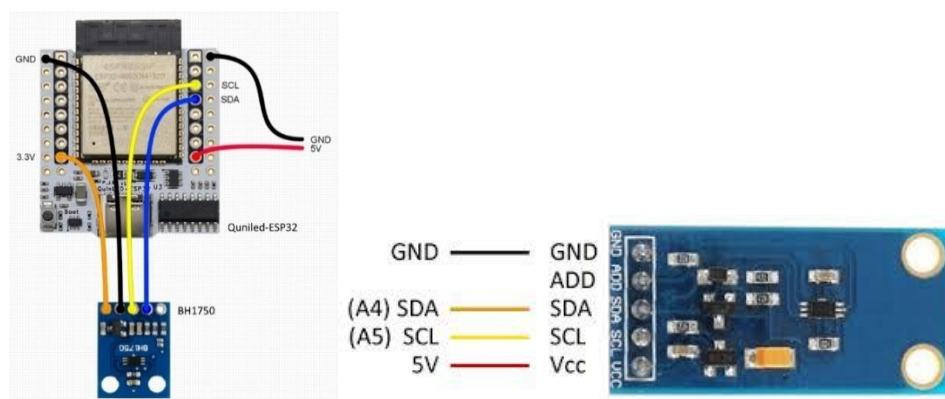
Figura 22 Sensor DHT11 conectado con ESP32



Fuente: Elaborado por el Autor

3.1.3. Conexión del sensor de intensidad de luz (BH1750) con la placa ESP32

Figura 23 Sensor BH1750 conectado con placa ESP32



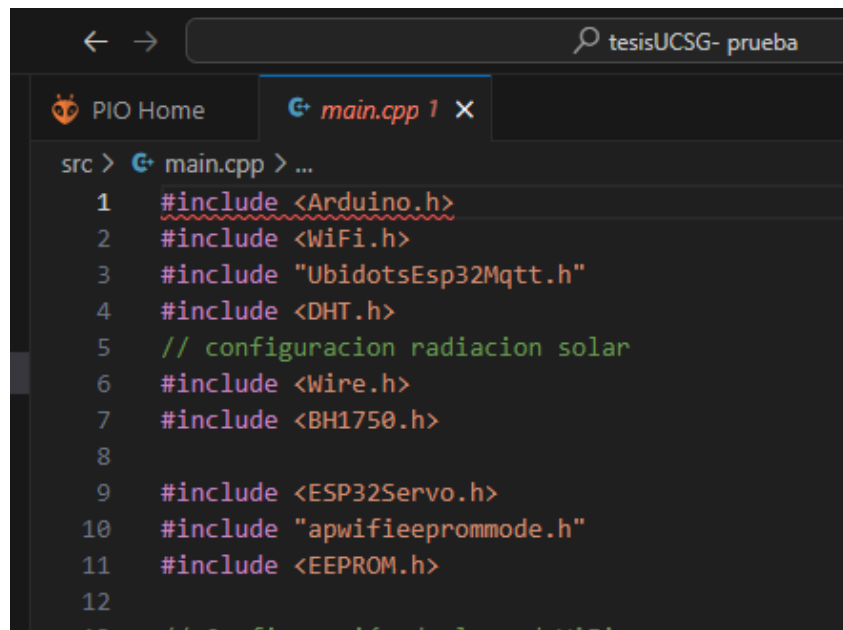
Fuente: Elaborado por el Autor

3.2. Desarrollo del código

Para el desarrollo del código se utilizará el algoritmo que consiste en colocar las librerías que se van a usar, declarar las variables, presentar texto y valores en la pantalla y presentar también la señal de los pines de cada sensor que está conectado al ESP32.

Se incluye archivos de encabezado en el programa. Esto permite que el programa funcione con bibliotecas o funcionalidades específicas.

Figura 24 Librerías utilizadas en el desarrollo del código

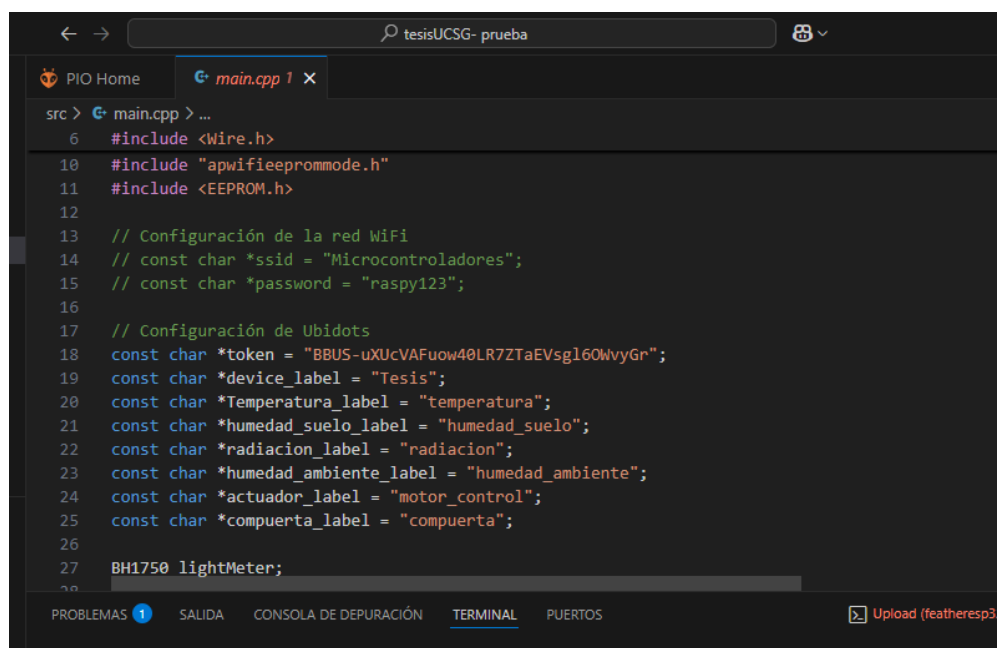


```
src > main.cpp > ...
1  #include <Arduino.h>
2  #include <WiFi.h>
3  #include "UbidotsEsp32Mqtt.h"
4  #include <DHT.h>
5  // configuracion radiacion solar
6  #include <Wire.h>
7  #include <BH1750.h>
8
9  #include <ESP32Servo.h>
10 #include "apwifleeprommode.h"
11 #include <EEPROM.h>
12
```

Elaborado por: El Autor

A continuación, se configura la red wifi, la plataforma Ubidots. Se declara las funciones que se van a usar como lo muestra la figura 24.

Figura 25 Declaración de funciones y configuración de red wifi

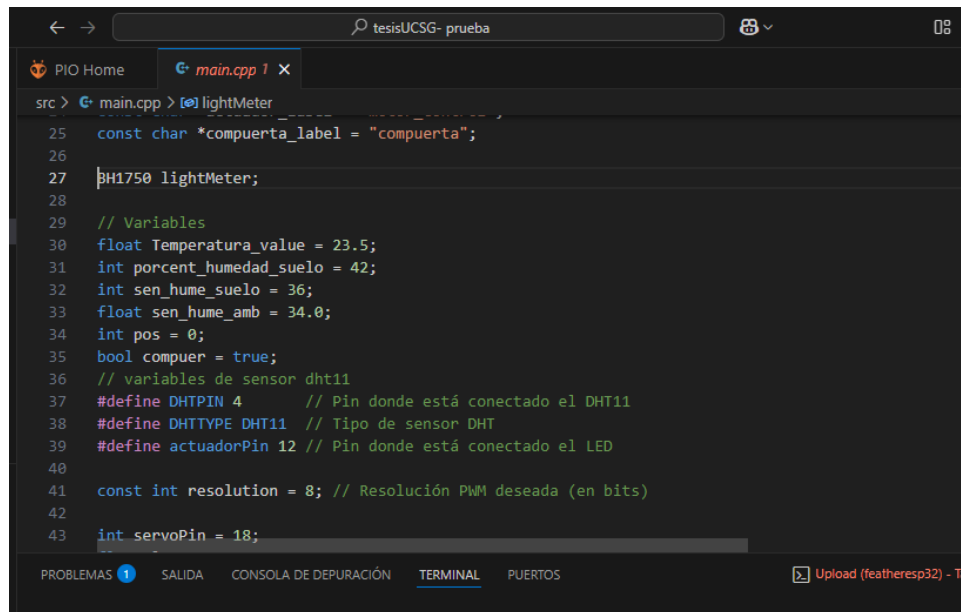


```
src > main.cpp > ...
6  #include <Wire.h>
10 #include "apwifleeprommode.h"
11 #include <EEPROM.h>
12
13 // Configuración de la red WiFi
14 // const char *ssid = "Microcontroladores";
15 // const char *password = "raspy123";
16
17 // Configuración de Ubidots
18 const char *token = "BBUS-uXUcVAFuow40LR7ZTaEVsgl60WvyGr";
19 const char *device_label = "Tesis";
20 const char *Temperatura_label = "temperatura";
21 const char *humedad_suelo_label = "humedad_suelo";
22 const char *radiacion_label = "radiacion";
23 const char *humedad_ambiente_label = "humedad_ambiente";
24 const char *actuador_label = "motor_control";
25 const char *compuerta_label = "compuerta";
26
27 BH1750 lightMeter;
```

Elaborado por: El Autor

Luego procedemos a almacenar números decimales o de punto flotante, también enteros que es el rango en el cual se deben mantener las variables climáticas de los huertos urbanos como indica la figura 25.

Figura 26 Almacenamiento de números de punto flotante y enteros utilizados en las variables climáticas

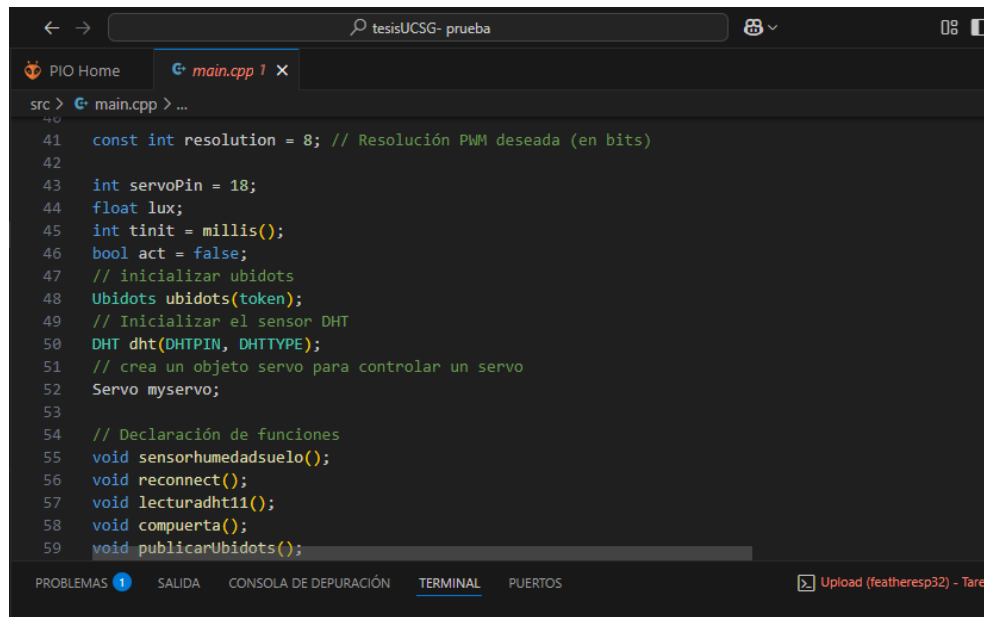


```
src > main.cpp > lightMeter
25 const char *compuerta_label = "compuerta";
26
27 BH1750 lightMeter;
28
29 // Variables
30 float Temperatura_value = 23.5;
31 int porcent_humedad_suelo = 42;
32 int sen_hume_suelo = 36;
33 float sen_hume_amb = 34.0;
34 int pos = 0;
35 bool compuer = true;
36 // variables de sensor dht11
37 #define DHTPIN 4 // Pin donde está conectado el DHT11
38 #define DHTTYPE DHT11 // Tipo de sensor DHT
39 #define actuadorPin 12 // Pin donde está conectado el LED
40
41 const int resolution = 8; // Resolución PWM deseada (en bits)
42
43 int servoPin = 18;
```

Elaborado por: El Autor

Ahora se procede a inicializar ubidots y el sensor DTH. Se crea el objeto servo para controlar el servo motor y se hace declaración de funciones como indica la figura 26.

Figura 27 *Desarrollo de código*

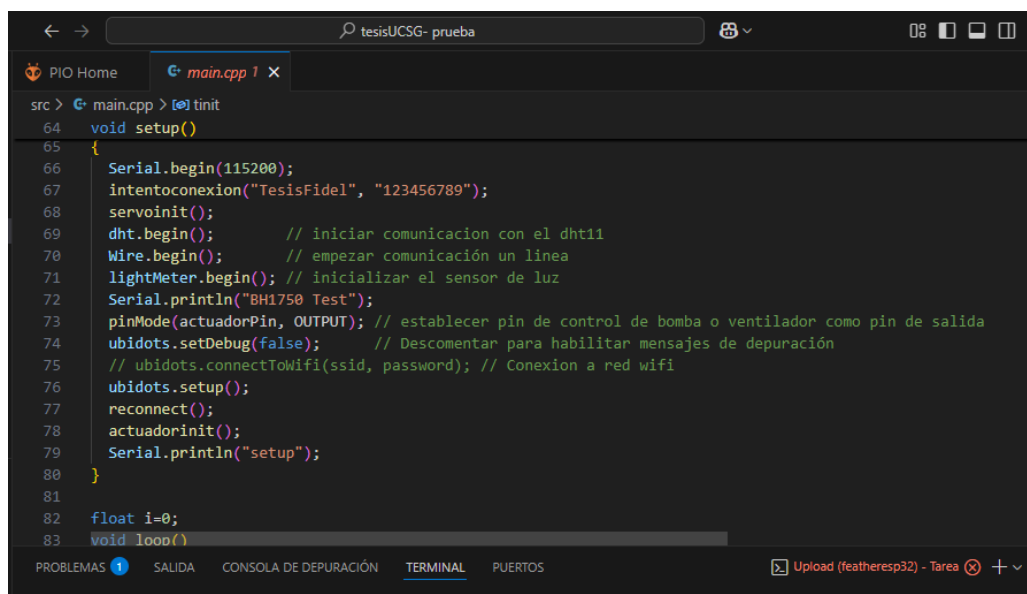


```
src > main.cpp > ...
41  const int resolution = 8; // Resolución PWM deseada (en bits)
42
43  int servoPin = 18;
44  float lux;
45  int tinit = millis();
46  bool act = false;
47  // inicializar ubidots
48  Ubidots ubidots(token);
49  // Inicializar el sensor DHT
50  DHT dht(DHTPIN, DHTTYPE);
51  // crea un objeto servo para controlar un servo
52  Servo myservo;
53
54  // Declaración de funciones
55  void sensorhumedadsuelo();
56  void reconnect();
57  void lecturadht11();
58  void compuerta();
59  void publicarUbidots();
```

Elaborado por: El Autor

Se abre un puerto serie y establece la velocidad de transmisión de datos en bits por segundo (115200 baudios). Se inicia comunicación con dht11, con sensor de luz, se establece pin de control de bomba o ventilador y conexión a red wifi. Como se muestra en figura 27.

Figura 28 *Desarrolla el código*

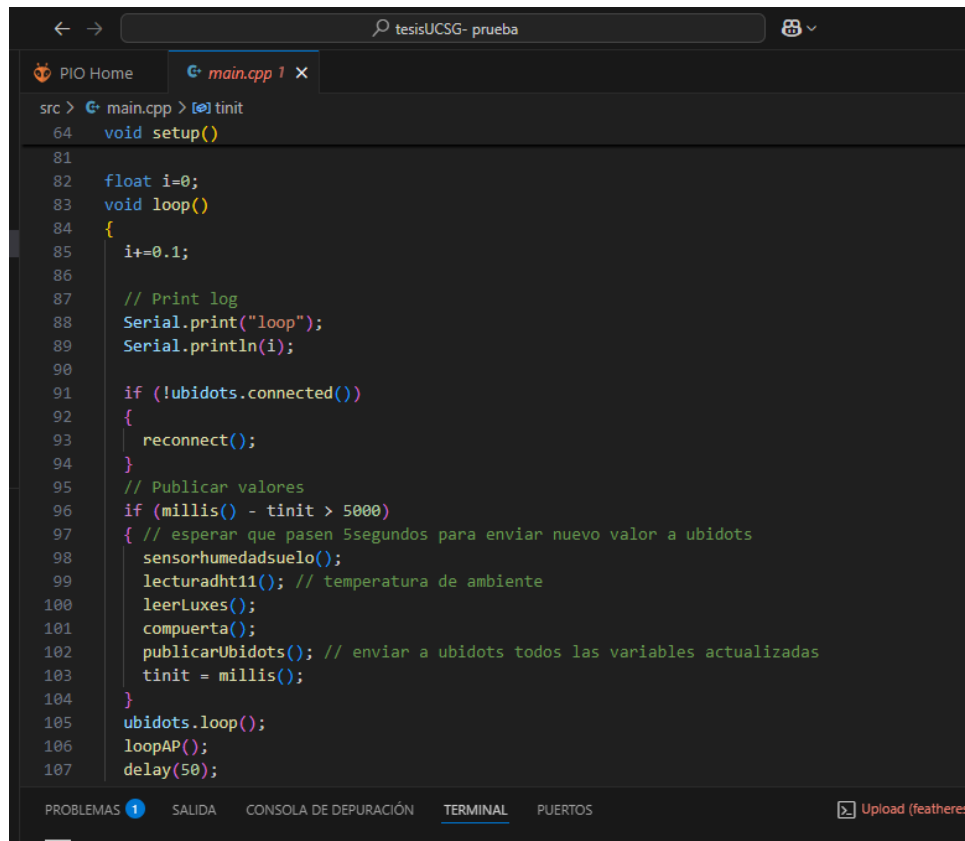


```
src > main.cpp > tinit
64  void setup()
65  {
66      Serial.begin(115200);
67      intentoconexion("TesisFidel", "123456789");
68      servoinit();
69      dht.begin(); // iniciar comunicacion con el dht11
70      Wire.begin(); // empezar comunicación un línea
71      lightMeter.begin(); // inicializar el sensor de luz
72      Serial.println("BH1750 Test");
73      pinMode(actuadorPin, OUTPUT); // establecer pin de control de bomba o ventilador como pin de salida
74      ubidots.setDebug(false); // Descomentar para habilitar mensajes de depuración
75      // ubidots.connectToWifi(ssid, password); // Conexion a red wifi
76      ubidots.setup();
77      reconnect();
78      actuadorinit();
79      Serial.println("setup");
80  }
81
82  float i=0;
83  void loop()
```

Elaborado por: El Autor

Se realiza la configuración para que espere cinco para enviar nuevo valor a ubidots y luego se envían todas las variables actualizadas. Como muestra la figura 28.

Figura 29 Desarrolla el código

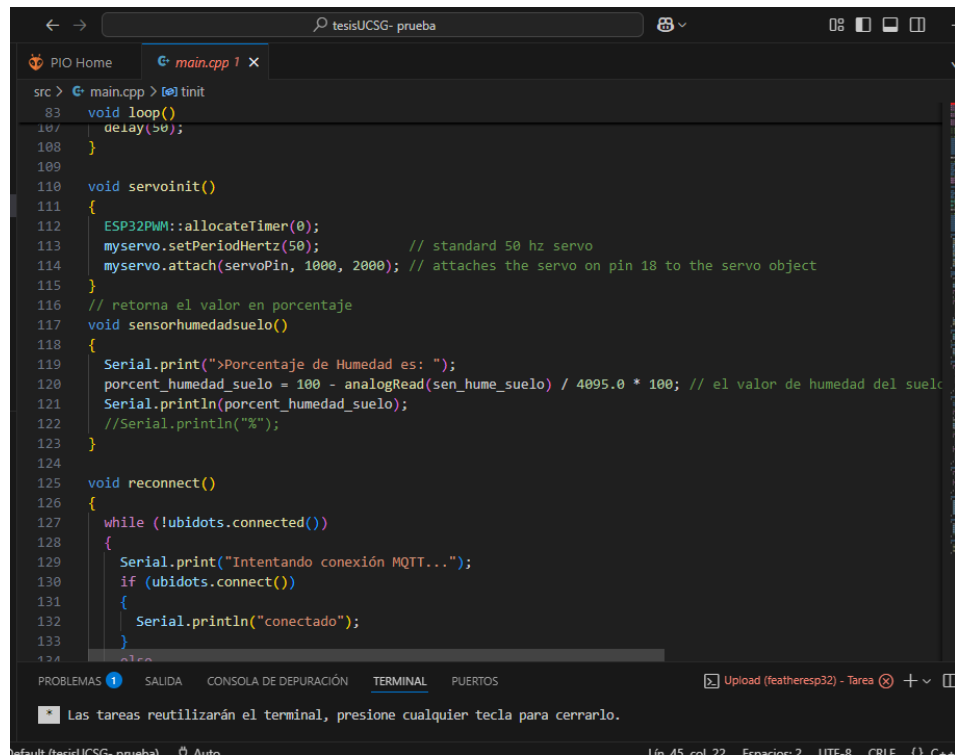


```
src > main.cpp > tinit
64 void setup()
81
82 float i=0;
83 void loop()
84 {
85     i+=0.1;
86
87     // Print log
88     Serial.print("loop");
89     Serial.println(i);
90
91     if (!ubidots.connected())
92     {
93         reconnect();
94     }
95     // Publicar valores
96     if (millis() - tinit > 5000)
97     { // esperar que pasen 5 segundos para enviar nuevo valor a ubidots
98         sensorhumedadsuelo();
99         lecturadht11(); // temperatura de ambiente
100         leerLuxes();
101         compuerta();
102         publicarUbidots(); // enviar a ubidots todos las variables actualizadas
103         tinit = millis();
104     }
105     ubidots.loop();
106     loopAP();
107     delay(50);
```

Elaborado por: El Autor

Se realiza la configuración para que el valor del sensor de humedad del suelo retorne en cantidad de porcentaje. Como se muestra en la figura 29.

Figura 30 Desarrolla el código

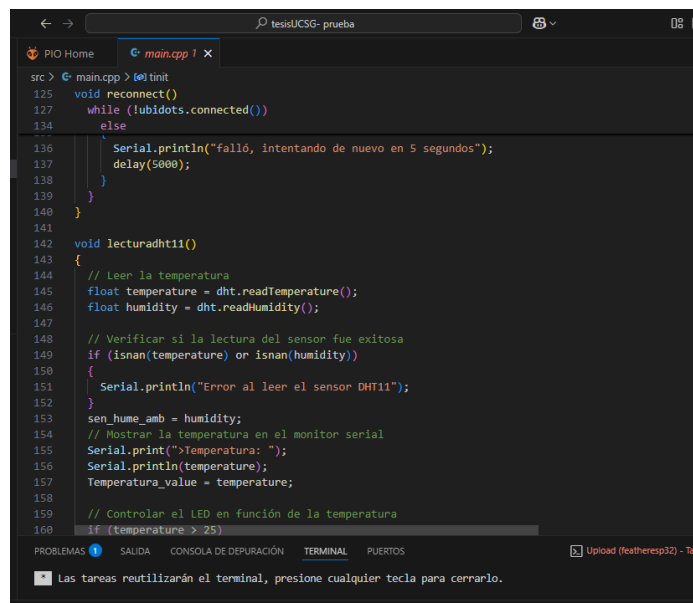


```
src > main.cpp > tinit
83 void loop()
107   delay(50);
108 }
109
110 void servoinit()
111 {
112   ESP32PWM::allocateTimer(0);
113   myservo.setPeriodHertz(50); // standard 50 hz servo
114   myservo.attach(servoPin, 1000, 2000); // attaches the servo on pin 18 to the servo object
115 }
116 // retorna el valor en porcentaje
117 void sensorhumedad_suelo()
118 {
119   Serial.print(">Porcentaje de Humedad es: ");
120   percent_humedad_suelo = 100 - analogRead(sen_hume_suelo) / 4095.0 * 100; // el valor de humedad del suelo
121   Serial.println(percent_humedad_suelo);
122   //Serial.println("%");
123 }
124
125 void reconnect()
126 {
127   while (!ubidots.connected())
128   {
129     Serial.print("Intentando conexión MQTT...");
130     if (ubidots.connect())
131     {
132       Serial.println("conectado");
133     }
134   }
135 }
```

Elaborado por: El Autor

Se realiza la configuración para leer la temperatura y la humedad del sensor dht,ademas se verifica si la lectura del sensor fue exitosa y se muestra en el monitor serial. Como se muestra en la figura 30.

Figura 31 Desarrolla el código



```
src > main.cpp > tinit
125 void reconnect()
127   while (!ubidots.connected())
134   else
136   {
137     Serial.println("falló, intentando de nuevo en 5 segundos");
138     delay(5000);
139   }
140 }
141
142 void lecturaDHT11()
143 {
144   // Leer la temperatura
145   float temperature = dht.readTemperature();
146   float humidity = dht.readHumidity();
147
148   // Verificar si la lectura del sensor fue exitosa
149   if (isnan(temperature) or isnan(humidity))
150   {
151     Serial.println("Error al leer el sensor DHT11");
152   }
153   sen_hume_amb = humidity;
154   // Mostrar la temperatura en el monitor serial
155   Serial.print(">Temperatura: ");
156   Serial.println(temperature);
157   Temperatura_value = temperature;
158
159   // Controlar el LED en función de la temperatura
160   if (temperature > 25)
```

Elaborado por: El Autor

Se realiza la configuración para encender el LED si la temperatura es mayor a

35°C o sino se apaga el LED si la temperatura está entre 18°C y 35°C. Como se muestra en la figura 31.

Figura 32 Desarrolla el código

```

src > main.cpp > tinit
142 void lecturaDHT11()
143 {
144     if (temperature > 25)
145     {
146         ledcWrite(2, int(temperature*4)); // Encender el LED si la temperatura es mayor a 35°C
147         act = true;
148         Serial.print(">Temperatura alta:");
149         Serial.println(act);
150     }
151     else if (temperature >= 18 && temperature <= 35)
152     {
153         ledcWrite(2, 0); // Apagar el LED si la temperatura está entre 18°C y 35°C
154         act = false;
155         Serial.print(">Temperatura normal:");
156         Serial.println(act);
157     }
158 }
159
160 void compuerta()
161 {
162     if (lux > 60)
163     {
164         myservo.write(60);
165         compuerta = true;
166     }
167     else
168     {
169         myservo.write(10);
170         compuerta = false;
171     }
172 }

```

Elaborado por: El Autor

Se realiza la configuración para enviar y recibir datos de la plataforma ubidots y para actualizar. Luego se envían los datos a través del puerto serial. Como se muestra en la figura 32.

Figura 33 Desarrolla el código

```

src > main.cpp > lecturaDHT11
176 void compuerta()
177 {
178 }
179
180 void publicarUbidots()
181 {
182     ubidots.add(Temperatura_label, Temperatura_value); // temperatura ambiente del dht11
183     ubidots.add(humedad_ambiente_label, sen_hume_amb); // humedad ambiente del dht11
184     ubidots.add(radiacion_label, float(lux * 0.0079)); // radiacion solar en W/m2
185     ubidots.add(actuador_label, act); // se actualiza el valor
186     ubidots.add(compuerta_label, compuerta); // se actualiza el valor de compuerta
187     ubidots.add(humedad_suelo_label, percent_humedad_suelo); // porcentaje de humedad suelo si está seco 0%
188     ubidots.publish(device_label);
189 }
190
191 void leerLuxes()
192 {
193     lux = lightMeter.readLightLevel(); // cantidad de luz
194     Serial.print(">light: ");
195     Serial.println(lux);
196     Serial.print(">radiacion:");
197     Serial.println(lux * 0.0079);
198     //Serial.println(" w/m2");
199 }
200
201 void actuadorinit()
202 {
203     // Configurar la resolución del PWM
204     ledcSetup(2, 5000, resolution); // Canal 0, frecuencia de 5 kHz
205     // Asignar el pin al canal PWM
206     ledcAttachPin(actuadorPin, 2);
207 }

```

Elaborado por: El Autor

3.3. Código Funcionando

Se presenta el código desarrollado y trabajando los sensores con la placa ESP32
`#include <Arduino.h>`

```

#include <WiFi.h>
#include "UbidotsEsp32Mqtt.h"
#include <DHT.h>
// configuracion radiacion solar
#include <Wire.h>
#include <BH1750.h>

#include <ESP32Servo.h>
#include "apwifieeprommode.h"
#include <EEPROM.h>

// Configuración de la red WiFi
// const char *ssid = "Microcontroladores";
// const char *password = "raspy123";

// Configuración de Ubidots
const char *token = "BBUS-uXUcVAFuow40LR7ZTaEVsgl6OWvyGr";
const char *device_label = "Tesis";
const char *Temperatura_label = "temperatura";
const char *humedad_suelo_label = "humedad_suelo";
const char *radiacion_label = "radiacion";
const char *humedad_ambiente_label = "humedad_ambiente";
const char *actuador_label = "motor_control";
const char *compuerta_label = "compuerta";

BH1750 lightMeter;

// Variables
float Temperatura_value = 23.5;
int percent_humedad_suelo = 42;
int sen_hume_suelo = 36;
float sen_hume_amb = 34.0;
int pos = 0;
bool compuer = true;
// variables de sensor dht11
#define DHTPIN 4    // Pin donde está conectado el DHT11

```

```

#define DHTTYPE DHT11 // Tipo de sensor DHT
#define actuadorPin 12 // Pin donde está conectado el LED

const int resolution = 8; // Resolución PWM deseada (en bits)

int servoPin = 18;
float lux;
int tinit = millis();
bool act = false;
// inicializar ubidots
Ubidots ubidots(token);
// Inicializar el sensor DHT
DHT dht(DHTPIN, DHTTYPE);
// crea un objeto servo para controlar un servo
Servo myservo;

// Declaración de funciones
void sensorhumedadsuelo();
void reconnect();
void lecturadht11();
void compuerta();
void publicarUbidots();
void leerLuxes();
void servoinit();
void actuadorinit();

void setup()
{
  Serial.begin(115200);
  intentoconexion("TesisFidel", "123456789");
  servoinit();
  dht.begin(); // iniciar comunicacion con el dht11
  Wire.begin(); // empezar comunicación un linea
  lightMeter.begin(); // inicializar el sensor de luz
  Serial.println("BH1750 Test");
}

```

```

    pinMode(actuadorPin, OUTPUT); // establecer pin de control de bomba o
    ventilador como pin de salida
    ubidots.setDebug(false);    // Descomentar para habilitar mensajes de depuración
    // ubidots.connectToWifi(ssid, password); // Conexion a red wifi
    ubidots.setup();
    reconnect();
    actuadorinit();
    Serial.println("setup");
}

float i=0;
void loop()
{
    i+=0.1;

    // Print log
    Serial.print("loop");
    Serial.println(i);

    if (!ubidots.connected())
    {
        reconnect();
    }
    // Publicar valores
    if (millis() - tinit > 5000)
    { // esperar que pasen 5 segundos para enviar nuevo valor a ubidots
        sensorhumedadsuelo();
        lecturadht11(); // temperatura de ambiente
        leerLuxes();
        compuerta();
        publicarUbidots(); // enviar a ubidots todos las variables actualizadas
        tinit = millis();
    }
    ubidots.loop();
    loopAP();
    delay(50);
}

```

```

}

void servoinit()
{
  ESP32PWM::allocateTimer(0);
  myservo.setPeriodHertz(50);      // standard 50 hz servo
  myservo.attach(servoPin, 1000, 2000); // attaches the servo on pin 18 to the servo
  object
}
// retorna el valor en porcentaje
void sensorhumedadsuelo()
{
  Serial.print(">Porcentaje de Humedad es: ");
  percent_humedad_suelo = 100 - analogRead(sen_hume_suelo) / 4095.0 * 100; //
  el valor de humedad del suelo se guarda en la varianble percent_humedad_suelo
  Serial.println(percent_humedad_suelo);
  //Serial.println("%");
}

void reconnect()
{
  while (!ubidots.connected())
  {
    Serial.print("Intentando conexión MQTT...");
    if (ubidots.connect())
    {
      Serial.println("conectado");
    }
    else
    {
      Serial.println("falló, intentando de nuevo en 5 segundos");
      delay(5000);
    }
  }
}

```

```

void lecturadht11()
{
    // Leer la temperatura
    float temperature = dht.readTemperature();
    float humidity = dht.readHumidity();

    // Verificar si la lectura del sensor fue exitosa
    if (isnan(temperature) or isnan(humidity))
    {
        Serial.println("Error al leer el sensor DHT11");
    }
    sen_hume_amb = humidity;
    // Mostrar la temperatura en el monitor serial
    Serial.print(">Temperatura: ");
    Serial.println(temperature);
    Temperatura_value = temperature;

    // Controlar el LED en función de la temperatura
    if (temperature > 25)
    {
        ledcWrite(2, int(temperature*4)); // Encender el LED si la temperatura es mayor a
35°C
        act = true;
        Serial.print(">Temperatura alta:");
        Serial.println(act);
    }
    else if (temperature >= 18 && temperature <= 35)
    {
        ledcWrite(2, 0); // Apagar el LED si la temperatura está entre 18°C y 35°C
        act = false;
        Serial.print(">Temperatura normal:");
        Serial.println(act);
    }
}

void compuerta()

```

```

{
  if (lux > 60)
  {
    myservo.write(60);
    compuer = true;
  }
  else
  {
    myservo.write(10);
    compuer = false;
  }
}

void publicarUbidots()
{
  ubidots.add(Temperatura_label, Temperatura_value);      // temperatura
  ambiente del dht11
  ubidots.add(humedad_ambiente_label, sen_hume_amb);      // humedad
  ambiente del dht11
  ubidots.add(radiacion_label, float(lux * 0.0079));      // radiacion solar en W/m2
  ubidots.add(actuador_label, act);      // se actualiza el valor
  ubidots.add(compuerta_label, compuer);      // se actualiza el valor de compuerta
  ubidots.add(humedad_suelo_label, percent_humedad_suelo); // porcentaje de
  humedad suelo si está seco 0%
  ubidots.publish(device_label);
}

void leerLuxes()
{
  lux = lightMeter.readLightLevel(); // cantidad de luz
  Serial.print(">Light: ");
  Serial.println(lux);
  Serial.print(">radiacion:");
  Serial.println(lux * 0.0079);
  //Serial.println(" w/m2");
}

```

```

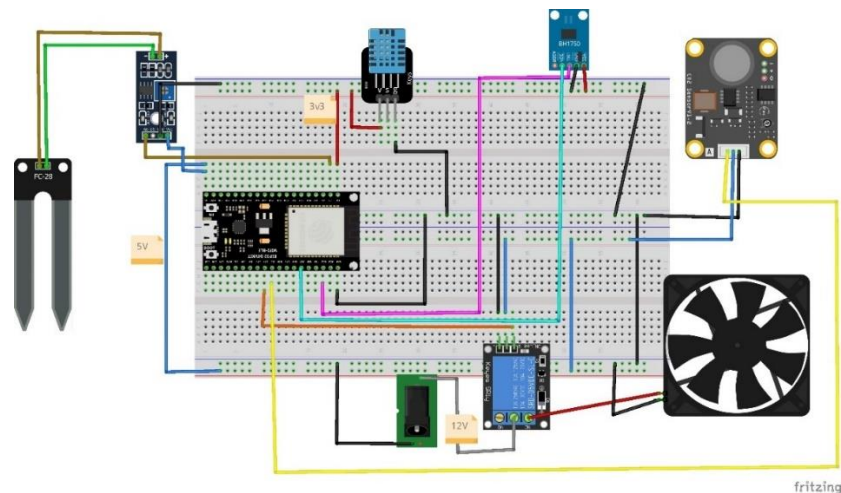
void actuadorinit()
{
    // Configurar la resolución del PWM
    ledcSetup(2, 5000, resolution); // Canal 0, frecuencia de 5 kHz
    // Asignar el pin al canal PWM
    ledcAttachPin(actuadorPin, 2);
}

```

3.4. Prueba en Sitio

Se realiza la conexión con los sensores de temperatura y humedad, sensor de humedad del suelo, sensor de intensidad de luz que controla los rayos UV en protoboard y conectados en los terminales de la placa ESP32, tal como vemos en la figura 34.

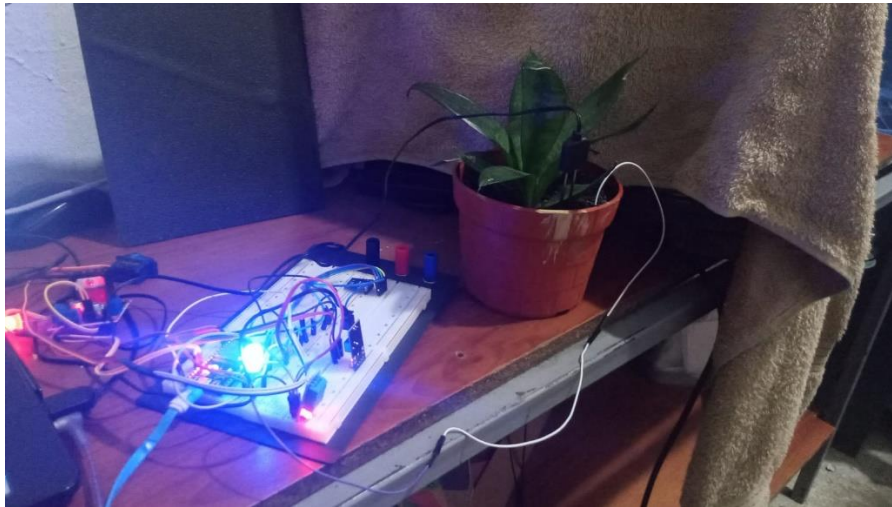
Figura 34 Conexión de ESP 32 con los sensores de variables climáticas



Elaborado por: El Autor

Se realiza pruebas para medir los valores de sensor de humedad y de temperatura colocados en la planta. Como se muestra en figura 34.

Figura 35 Conexiones realizadas en protoboard y sensores ubicados en planta



Elaborado por: El Autor

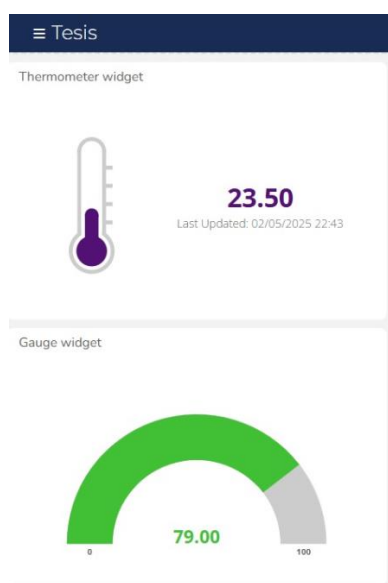
El sensor de humedad de la tierra, funcionando perfectamente tal como muestra el monitor serie de IDE Arduino

Figura 36 Muestra del sensor de humedad del suelo en monitor serie del IDE de Arduino

```
Porcentaje de Humedad es: 100%
Porcentaje de Humedad es: 82%
Porcentaje de Humedad es: 83%
Porcentaje de Humedad es: 73%
Porcentaje de Humedad es: 63%
Porcentaje de Humedad es: 0%
Porcentaje de Humedad es: 77%
Porcentaje de Humedad es: 84%
Porcentaje de Humedad es: 98%
Porcentaje de Humedad es: 69%
Porcentaje de Humedad es: 100%
```

Elaborado por: El autor

Figura 37 *Medición del Sensor de Humedad en Ubidots*



Elaborado por: El autor

Luego de realizar lecturas y pruebas en la plataforma se determinó, que esta realiza una muestra cada cinco segundos dependiendo de la velocidad del enlace, esto se debe a que Ubidots utiliza el protocolo MQTT, diseñado originalmente para sensores y dispositivos que puedan soportar cierta latencia en la red.

3.4. Análisis de Costos

En la Tabla 3.5 se indican los costos unitarios de cada elemento y el costo total para el prototipo que ha sido propuesto para el desarrollo este proyecto.

Tabla 2 *Costos de los componentes*

TARJETA	COSTO POR UNIDAD
DTH22	6
CABLES Y CONECTORES	4
NodeMCU ESP32	8

Sensor de Humedad del suelo	9
Sensor de Radiación Solar	7,50
Ventilador	2
PROGRAMACION	350
TOTAL	386,5

Fuente: Elaborado por el autor

Se debe contar con una conexión Wi-Fi y servicio de proveedor de Internet que no es parte del costo del prototipo.

3.5. Discusión y análisis de los resultados

En la presente investigación se determinó que las mediciones realizadas por los sensores tienen una característica representativa, las cuales se las toma cada cinco segundos, esto no es una situación crítica ya que no se busca la reproducción exacta de la señal, sino más bien valores discretos, que se refresquen cada cierto tiempo.

En lo que respecta a la investigación de los dispositivos en el mercado local, estos son económicos que para un proyecto básico es perfectamente viable y factible, esto es importante pues se pueden aplicar en el desarrollo de huertos en donde exista un clima controlado, como una alternativa a los grandes cultivos que necesita mucha inversión inicial.

El inconveniente relacionado con la configuración de Ubidots es que se debe cambiar el nombre de la red SSID en el código, ya que en el mismo es donde se debe asignar el nombre de la red.

Conclusiones

1. Se desarrolló un código eficiente en C++ para la adquisición de datos de temperatura, humedad y radiación solar, utilizando sensores adecuados para las condiciones de los huertos urbanos en Guayaquil.
2. Se diseñó una arquitectura de sistema que integre los sensores, la placa ESP32 y la plataforma Ubidots, garantizando una comunicación estable y la transmisión de datos en tiempo real.
3. Se desarrolló una interfaz de usuario intuitiva en Ubidots para la visualización y el análisis de los datos climáticos, que permita a los usuarios generar reportes y gráficos personalizados.
4. Se calibró los sensores utilizados, pero no se logra valores exactos de la señal sino más bien valores discretos y aproximados para asegurar la confiabilidad de los datos recolectados en un rango de condiciones climáticas típicas de la región.

Recomendaciones

1. Se recomienda para futuras investigaciones implementar más sensores para tener una amplia información de los estados del tiempo
2. Utilizar el prototipo en huertos o plantaciones y sirve como modelo educativo para esquematizar las partes del mismo en el aula.
3. Se recomienda conectar el dispositivo a actuadores como bombas para poder controlar la humedad en caso que sea necesario dentro de plantaciones o huertos.

REFERENCIAS

- Abrantes, P. (2020). La educacion en tiempos virales. caso de Portugal. RASE , 30-43.
- AEA. (02 de 04 de 2021). Asociacion educacion abierta. Recuperado el 10 de 06 de 2021, de <https://educacionabierta.org/>
- AEMET. (04 de 08 de 2020). Obtenido de <https://enviraiot.es/que-son-sensores-meteorologicos/>
- Aguirre, C., Agüero, J., Félix, E., Landeo, J., & Wong, A. (2019). Collares geolocalizadores para mascotas vinculados a una aplicación móvil. Obtenido de Repositorio Académico Universidad Peruana de Ciencias Aplicadas: <http://hdl.handle.net/10757/626553>
- Ahson, S., & Ilyas, M. (2008). RFID handbook : applications, technology, security, and privacy. CRC Press.
- Arduino, S. o. (s.f.). Arduino. Recuperado el 12 de 07 de 2021, de <https://arduino.cl/que-es-arduino/>
- Association for the Advancemnte of Medical Instrumentation. (s.f.). Recuperado el 10 de 07 de 2021, de AAMI: <https://www.ami.org>
- Basantes, J. (2016). Analisis de factibilidad técnica y de viabilidad cmercial de dispositivos para localización de mascotas caninas mediante el uso de tecnología en el Distrito Metropolitano de Quito. Obtenido de Repositorio Pontificia Universidad Católica del Ecuador: [file:///C:/Users/MINEDUC/Downloads/TESIS%20BASANTES%20FELIPE%20\(1\).pdf](file:///C:/Users/MINEDUC/Downloads/TESIS%20BASANTES%20FELIPE%20(1).pdf)
- Bistel , R., Fajardo , A., & Valdes, E. (2013). Diseño de un instrumento virtual para la calibraicon de Sistemas Electrocardiograficos. La Habana: scielo.
- Blasco, N. (2017). EndoG influye en el crecimiento y la proliferación celular con impacto en el desarrollode tejidos y la progresión tumora. Barcelona: Universidad de Lleida.
- Bonal, X., & Gonzalez, S. (2020). Educacion formal e informal en confinamiento. una creciente desigualdad de oportunidades de aprendizaje. RASE, 44-62.

- CAF. (25 de Febrero de 2020). CAF. Recuperado el 4 de Mayo de 2021, de <https://www.caf.com/es/conocimiento/visiones/2020/02/transformacion-digital-para-la-america-latina-del-s21/>
- Canna. (05 de 06 de 2021). Influencia de la temperatura ambiental de las plantas. Obtenido de <https://www.canna.es/user/dealersearch>
- Castillo, E. (2018). Diseño de una etiqueta pasiva sin chip para aplicaciones RFID en UWB. Guayaquil: UCSG.
- Certificación ISO 13485—Gestión de productos sanitarios | NQA. (s. f.). Recuperado 14 de abril de 2025, de <https://www.nqa.com/es-mx/certification/standards/iso-13485>
- Chan, M. (2008). La atención primaria de salud, más necesaria que nunca OMS. Recuperado el 16 de 7 de 2021, de https://apps.who.int/iris/bitstream/handle/10665/43952/9789243563732_spa.pdf?sequence=1&isAllowed=y
- Chandra Karmakar, N. (2010). Handbook of smart antennas for RFID systems. John Wiley & Sons.
- Costa, F., Genovesi, S., & Monorchio, A. (2013). A Chipless RFID Based on Multiresonant High-Impedance Surfaces. IEEE TRANSACTIONS ON MICROWAVE THEORY AND TECHNIQUES, 61(1).
- Echeita, G. (2020). La Pandemia del Covid19. Una oportunidad para pensar como hacer más inclusivos nuestros sistemas educativos. Madrid: Universidad Autónoma de Madrid.
- Edel, R. (2018). El rendimiento académico, concepto, investigación y desarrollo. Revista americana sobre calidad, eficacia y cambio en la educación.
- edeptec. (2 de 7 de 2021). edptec. Obtenido de <https://www.edeptec.com/2021/02/que-es-arduino-funcionamiento-y-partes.html>
- electrodrone. (3 de 4 de 2020). electrodrone. Obtenido de <https://store.electrodrone.net/index.php/producto/sensor-de-humedad-y-temperatura-dht22/>
- electronic, M. (05 de 06 de 2020). que es arduino. Obtenido de <https://arduino.cl/que-es-arduino/>

- Envira. (05 de 06 de 2020). Obtenido de <https://enviraiot.es/que-son-sensores-meteorologicos/>
- Fajuri, A. (2009). Manual de arritmias. Recuperado el 12 de 07 de 2021, de manual de arritmias: https://medicina.uc.cl/wp-content/uploads/2018/06/Arritmias_1.pdf
- FECYT. (2014). Obtenido de <https://www.fecyt.es/es/publicacion/unidad-didactica-meteorologia-y-climatologia>
- Ferrer, R. (2017). Desarrollo de una estacion meteorologica de bajo costo. Bucaramanga: U. Santo Tomas.
- Gomez, M. (4 de abril de 2018). MARCO TEORICO. Recuperado el 4 de Mayo de 2021, de <https://www.marcoteorico.com/curso/50/definicion-de-marco-teorico>
- Gomez, Q., & Iglesias , A. (2004). Influencia del clima sobre la productividad agrícola en España. Global Environmental Change.
- Heras, A., las, F., Gómez, C., Franco, M. E., & Marzábal, M. (2009).
- IESALC. (2020). Covid 19 y educacion, los efectos inmediatos del dia despues. Caracas: IESALC.
- Jirasereeamornkol, J. W. (2005). Power Harvest Design for Semi-Passive UHF RFID Tag Using a Tunable Impedance Transformation. 9th Internatinal Symposium on Communications and Information Tech, 1441-1445.
- job, B. (3 de 4 de 2021). Bejob. Obtenido de <https://www.bejob.com/que-es-la-programacion-con-arduino-y-para-que-sirve/>
- libre, h. (05 de 06 de 2020). NodeMCU: la plataforma de IoT de código abierto. Obtenido de <https://www.hwlibre.com/nodemcu/>
- Llamas , L. (19 de 3 de 2013). Obtenido de <https://www.bolanosdj.com.ar/MOVIL/ARDUINO2/MedidorHumedadSuelo.pdf>
- Martinez, E. (2018). Tipo, alcance y diseño de la investigacion. Recuperado el 4 de mayo de 2021, de https://www.uaeh.edu.mx/docencia/VI_Lectura/maestria/documentos/LECT85.pdf

- Michelena, A. (19 de Octubre de 2020). Boletín de prensa 427. (MINTEL, Entrevistador)
- Moltoni, A., Irurueta, M., Negri, L., & Duro, S. (2010). Evaluación de collares para rastreo de animales basados en tecnología GPS. Obtenido de World Congress & Exhibition Engineering 2010-Argentina.: <https://inta.gob.ar/sites/default/files/script-tmp-collar.pdf>
- Munk, B. A. (2000). Frequency Selective Surfaces. Theory and Design. John Wiley & Sons.
- OMS. (2012). Evaluación de las necesidades de dispositivos médicos. Recuperado el 14 de 07 de 2021, de http://apps.who.int/iris/bitstream/handle/10665/44816/9789243501383_spa.pdf%3Bjsessionid%3D3F81F1D9F849671C224833C26F52F701?sequence=1
- Padron , A., & Dorta, A. (2020). Patogenia de las manifestaciones neurológicas asociadas con el SARS CoV2. Revista cubana de investigaciones biomédicas, 8.
- Peña, C. (2017). Arduino, de cero a experto. Buenos Aires: Six Ediciones.
- Pereira , G. (2014). Desarrollo de software embebido a través de transformación de modelos para aplicar en vigilancia de cultivos agrícolas. Medellín.
- Pereira de Siqueira Campos, A. (2008). Superfícies Seletivas em Frequência: análise e projeto. IFRN.
- Perez , S. (07 de 2018). Temas para la educación. Recuperado el 10 de 06 de 2021, de <https://www.feandalucia.ccoo.es/docu/p5sd7396.pdf>
- Preradovic, S., & Karmakar, N. C. (2012). Multiresonator-Based Chipless RFID. Barcode of the future. Springer.
- Ramirez, F. (2009). Fisiología cardíaca. Revista Médica, pag. 1.
- Roggiero, J. (2020). La ficción de educar a distancia. Revista de Sociología de la Educación, especial. COVID-19,: <http://dx.doi.org/10.7203/RASE.13.2.17126>, 172-182.
- Ruiz, J. (2013). Requerimientos agroecológicos de los cultivos. México: Prometeo Editores.

- Rujas, J., & Feito, R. (2020). La educacion en tiempos de pandemia. una situacion excepcional y cambiante. RASE, 4-13.
- Salamanca, U. (04 de 01 de 2021). dicciomed. Recuperado el 10 de 06 de 2021, de <https://dicciomed.usal.es/palabra/pandemia>
- Salud, M. d. (03 de 05 de 2021). MinSalud . Recuperado el 09 de 06 de 2021, de <https://www.salud.gob.ec/coronavirus-covid-19/>
- San José, J., Pastor, J., & García, A. (2012). RFID: La Identificación por Radiofrecuencia como futuro de la identificación de objetos. Obtenido de <https://www.researchgate.net/publication/275020704>
- San Mauro, M. (2013). Anatomia cardiaca, una manera intetral de estudiar las estrucutras del corazon y los grandes vasos. La Plata. Argentina: Universidad de la Plata. Editorial de la Univesidad de la Plata.
- Sánchez, R. (2014). ETIQUETA PASIVA DE RFID SIN CHIP PARA SENSADO DE MATERIALES. Tesis en opción al grado de Maestro en Ciencias en la especialidad de Electrónica., Instituto Nacional de Astrofísica, Óptica y Electrónica., Puebla. México.
- Santos, P. (2016). Diseño de una antena tag RFID pasiva de orden cero (ZOR) en UHF con metamateriales. Guayaquil: UCSG.
- Sanzi, I., Sainz , J., & Capilla, A. (2020). Efectos de la crisis del coronavirus en la eduacion. Madrid: OEI.
- Sofos. (4 de 8 de 2021). sofos organizacion del conocimiento. Obtenido de <http://www.sofoscorp.com/conoces-cuales-necesidades-cultivo/>
- Steduto, P., & Fereres, E. (2012). Respuesta del rendimiento de los cultivos del agua. Roma: ONU.
- Suarez, A. (26 de diciembre de 2020). La educacion durante la pandemia, evitar la desercion escolar. El telegrafo.
- Tarabini, A. (2020). Para que sirve la escuela. reflexiones sociologicas en tiempo de pandemia global. RASE, 145-155.
- Telectrónica. (2006). Introducción a la identificación por Radio Frecuencia-RFID.
- Ulloa, R., Ivankovich, G., & Yamazaki, M. (2020). Síndrome inflamatorio multisistémico asociado al Covid 19 en niños y adolescentes. Revista chilena de infectología.

UNESCO. (s.f.). Recuperado el 4 de mayo de 2021, de <https://es.unesco.org/covid19/educationresponse>

UNICEF. (19 de noviembre de 2020). Recuperado el 4 de mayo de 2021, de <https://www.unicef.org/ecuador/comunicados-prensa/unicef-pide-evitar-da%C3%B1os-irreversibles-en-educaci%C3%B3n-salud-nutrici%C3%B3n-y-bienestar>

UNICEF. (8 de Abril de 2020). UNICEF ORG. Recuperado el 4 de mayo de 2021, de <https://www.unicef.org/ecuador/comunicados-prensa/la-educaci%C3%B3n-debe-continuar-durante-la-emergencia-sanitaria-afirman-unesco-y>

urbana, E. (05 de 06 de 2020). Obtenido de https://educacionurbana.com/?page_id=302

Vega, G. (2012). Diseño y construcción de un electrocardiograma de 12 derivaciones para el análisis de señales cardíacas. Cuenca: UPS.



Presidencia
de la República
del Ecuador



Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

DECLARACIÓN Y AUTORIZACIÓN

Yo, **Llivicura Gordillo, Fidel Antonio** con C.C: # **0913267241** autor del trabajo de titulación: **Diseño de un sistema de monitorización de las variables climáticas en huertos urbanos de la costa utilizando plataforma abierta Arduino y Ubidots**, previo a la obtención del título de **Magister en Telecomunicaciones** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 11 de abril del **2025**

EL AUTOR

Llivicura Gordillo, Fidel Antonio

C.C: **0913267241**



**Presidencia
de la República
del Ecuador**



**Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes**



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA			
FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN			
TEMA Y SUBTEMA:	Diseño de un sistema de monitorización de las variables climáticas en huertos urbanos de la costa utilizando plataforma abierta Arduino y Ubidots		
AUTOR(ES)	Llivicura Gordillo, Fidel Antonio		
REVISOR(ES)/TUTOR(ES)	Bohórquez Heras, Daniel Bayardo; Ubilla González, Ricardo Xavier; Bohórquez Escobar, Celso Bayardo		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Sistema de Posgrado		
CARRERA:	Maestría en Telecomunicaciones		
TÍTULO OBTENIDO:	Magister en Telecomunicaciones		
FECHA DE PUBLICACIÓN:	11 de abril de 2025	No. DE PÁGINAS:	59 p.
ÁREAS TEMÁTICAS:	Agricultura de precisión, Tecnologías 4.0 para IoT, Dispositivos y sensores IoT, Protocolos de comunicación, Sistema de monitorización.		
PALABRAS CLAVES/KEYWORDS:	Cultivos, NodeMCUESP32, Arduino, Metodología, Ubidots, Automatización.		
RESUMEN/ABSTRACT: La investigación de este proyecto tiene como objetivos diseñar un sistema de bajo costo para monitorear las variables climáticas, desarrollar una interfaz intuitiva para visualizar los datos en tiempo real e implementar un sistema de alertas para prevenir problemas en los cultivos. La metodología incluye la conexión de sensores de variables climáticas a un microcontrolador NodeMCUESP32 a través de sus pines Vcc, GND y OUT, los cuales transmiten datos en tiempo real a la plataforma Ubidots mediante MQTT, que es un protocolo de mensajería que se utiliza para conectar dispositivos y aplicaciones a través de internet. Finalmente, en Ubidots se creará un dashboard dinámico y widgets a través de la configuración que permite nombrar todos los dispositivos climáticos que se usarán para visualizar, monitorear y analizar la información recolectada, permitiendo una gestión eficiente del cultivo y la reducción de desperdicios. Nuestro país se está abriendo paso hacia la tecnología 4.0 la misma que posee recursos como el internet de las cosas (IoT) , inteligencia artificial (IA), la automatización, el big data que son herramientas innovadoras para realizar procedimientos industriales en menos tiempo y con gran precisión. Es el caso de la Empresa Claro que destina los recursos de la Tecnología 4.0 para ofrecer servicios de alimentación inteligente llamada BIOFEEDER que consiste en mejorar la calidad de la producción de camarones. Otro ejemplo es la Empresa YARA que se encuentra establecida en nuestro país y ofrece agricultura de precisión orientado a las grandes empresas agrícolas. La contribución de este estudio sobresale en la optimización del Diseño de un sistema que integra hardware, software, conectividad, IoT y configuración de plataforma en la nube que da lugar a una mejor producción agrícola.			
ADJUNTO PDF:	☒ SI	☐ NO	
CONTACTO CON AUTOR/ES:	Teléfono: +	E-mail:	
CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):	Nombre: Bohórquez Escobar Celso Bayardo		
	Teléfono: 09995147293		
	E-mail: celso.bohorquez@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			