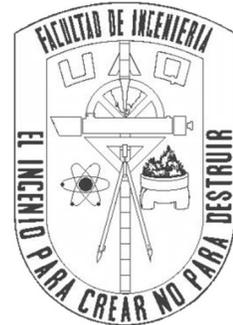


Universidad Autónoma de Querétaro



Facultad de Ingeniería
Campus San Juan del Río



Ingeniería Electromecánica

Desarrollo de un sistema de monitoreo meteorológico
de bajo costo basado en tecnología IoT.

PRESENTA:

Gustavo Medina Vega

Expediente:

272308

Dirigido por:

Dr. Luis Morales Velázquez

M.C. Ervin Galán Uribe

San Juan del Río, Querétaro, Noviembre, 2023



Dirección General de Bibliotecas y Servicios Digitales
de Información



Desarrollo de un sistema de monitoreo meteorológico
de bajo costo basado en tecnología IoT.

por

Gustavo Medina Vega

se distribuye bajo una [Licencia Creative Commons
Atribución-NoComercial-SinDerivadas 4.0
Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Clave RI: IGLIN-272308



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO

FACULTAD DE INGENIERÍA

CAMPUS SAN JUAN DEL RÍO



Desarrollo de un sistema de monitoreo meteorológico de bajo costo basado en tecnología IOT.

TESIS

Como parte de los requisitos para obtener el título de

INGENIERO ELECTROMECAÁNICO

Línea Terminal

Mecatrónica

Presenta

Gustavo Medina Vega

Dirigido por:

Dr. Luis Morales Velázquez

M.C. Ervin Galán Uribe

SINODALES:

Dr. Luis Morales Velázquez (6829)

Presidente

M.C. Ervin Galán Uribe

Secretario

Dr. Roque A. Osornio Ríos (5620)

Vocal

Dr. David A. Elvira Ortiz (16323)

Suplente

RESUMEN

En el contexto actual, donde el cambio climático está transformando los patrones climáticos globales, la necesidad de sistemas de vigilancia meteorológica confiables y accesibles es más crucial que nunca. Una solución ingeniosa y cada vez más popular es la estación meteorológica autónoma, capaz de recolectar información vital como la velocidad del viento, la temperatura y la humedad. Estas estaciones son esenciales tanto para el diseño de estrategias de adaptación al cambio climático como para comprender las condiciones climáticas actuales.

Este proyecto se enfoca en desarrollar una estación meteorológica autónoma y de bajo costo, que integra la tecnología del Internet de las Cosas (IoT) y la energía solar. Esta combinación permite un funcionamiento autónomo, minimizando la necesidad de mantenimiento y de intervención humana. Operando con energía solar, la estación puede ser instalada en zonas remotas o de difícil acceso, contribuyendo así a la sostenibilidad.

Cada componente ha sido seleccionado para asegurar la máxima eficiencia y rentabilidad. La implementación de la tecnología IoT en una estación meteorológica permite una recopilación de datos en tiempo real y un almacenamiento eficiente en la nube. Además, desarrollando una interfaz de usuario intuitiva ayuda a la visualización y el análisis de los datos meteorológicos. La precisión y fiabilidad de los datos recolectados se validan mediante comparaciones con estaciones meteorológicas establecidas.

Este proyecto ilustra cómo la ingeniería, especialmente en los campos de las energías renovables e IoT, puede ofrecer soluciones innovadoras y económicamente viables a los desafíos presentados por el cambio climático. La estación meteorológica diseñada en este estudio no solo proporciona un medio más eficiente de monitoreo climático, sino que también se perfila como un modelo adaptable y reconfigurable para aplicaciones futuras en diversos contextos y ubicaciones.

Palabras clave: Estación Meteorológica Autónoma, Tecnología IoT, Rentabilidad, Monitoreo Climático, Análisis de Datos.

ABSTRACT

In the current context, where climate change is transforming global climatic patterns, the need for reliable and accessible weather surveillance systems is more crucial than ever. An ingenious and increasingly popular solution is the autonomous weather station, capable of collecting vital information such as wind speed, temperature, and humidity. These stations are essential both for designing climate change adaptation strategies and for understanding current climatic conditions. This project focuses on developing a low-cost, autonomous weather station that integrates Internet of Things (IoT) technology and solar energy. This combination allows for autonomous functioning, minimizing the need for maintenance and human intervention.

Operating with solar energy, the station can be installed in remote or difficult-to-access areas, thus contributing to sustainability. Each component has been selected to ensure maximum efficiency and profitability. Implementing IoT technology in a weather station allows for real-time data collection and efficient cloud storage. Furthermore, developing an intuitive user interface aids in the visualization and analysis of meteorological data. The accuracy and reliability of the collected data are validated through comparisons with established weather stations.

This project illustrates how engineering, especially in the fields of renewable energies and IoT, can offer innovative and economically viable solutions to the challenges posed by climate change. The weather station designed in this study not only provides a more efficient means of climate monitoring but also emerges as an adaptable and reconfigurable model for future applications in various contexts and locations.

Keywords: Autonomous Weather Station, IoT Technology, Profitability, Climate Monitoring, Data Analysis.

*Dedicado a todos
aquellos que
iluminaron mi camino
académico y personal.*

Agradecimientos

Al culminar esta etapa de mi vida académica, me siento profundamente agradecido con todas aquellas personas que han sido pilares fundamentales en mi camino. Esta tesis no solo es el resultado de mi esfuerzo, sino también el reflejo del apoyo y la inspiración que he recibido de cada uno de ellos.

En primer lugar, mi más sincero agradecimiento al Dr. Luis Morales Velázquez, director de mi tesis. Su guía, paciencia y conocimientos han sido cruciales en cada fase de este proyecto. Sus consejos y su apoyo constante me han permitido enfrentar y superar los desafíos inherentes a esta investigación.

También quiero expresar mi gratitud hacia el M.C. Ervin Galán Uribe, cuya disposición para resolver dudas y brindar orientación ha sido invaluable. Su apoyo ha sido un factor determinante en el desarrollo de mi trabajo.

A mi familia, les debo un agradecimiento especial. Su amor, comprensión y apoyo incondicional han sido la base sobre la cual he construido mis sueños y aspiraciones. Gracias por estar siempre ahí, en cada paso de este viaje.

A mi querida Emilia, gracias por ser mi compañera de vida, por tu apoyo inagotable, tu paciencia y por creer en mí incluso cuando dudaba de mí mismo. Tu amor ha sido mi refugio y mi motivación.

No puedo dejar de mencionar a mis amigos: Constanza, Diana, Karla, Marco, Alan, Miguel, Josh, Antonio, Eduardo y aquellos que hoy no están con nosotros. Cada uno de ustedes ha aportado algo único a mi vida. Su amistad, consejos y momentos compartidos han sido una fuente de alegría y fortaleza.

Finalmente, quiero dedicar este trabajo a todos aquellos que han sido parte de mi viaje académico y personal. Cada conversación, cada palabra de aliento y cada gesto de apoyo han dejado una huella indeleble en mi vida.

Y, de manera muy especial, quiero recordar y agradecer a Rita, mi querido gato. Su presencia ha sido no solo una fuente de alegría y compañía, sino también una inspiración constante. Su recuerdo me motiva a seguir adelante en mi camino hacia nuevas investigaciones

Gracias a todos por ser parte de esta importante etapa de mi vida.

INDICE DE FIGURAS

Figura 1	Estación Meteorológica “Plutarco Elías Calles” (CONAGUA, 2005).....	11
Figura 2	Diagrama de bloques del Planteamiento General.	17
Figura 3	Diagrama de estación meteorológica (Obtenida de CONAGUA).	18
Figura 4	Conexión SPI (Autoría propia)	34
Figura 5	Conexión I2C (Autoría propia).....	35
Figura 6	Sensor de temperatura BME680.(autoría propia)	39
Figura 7	Diagrama lot (Autoría propia).	41
Figura 8	Diagrama de bloques de la Metodología propuesta.	47
Figura 9	Interfaz Web estación meteorológica.....	60
Figura 10	Tiempo de medición y transmisión por ciclo.....	62
Figura 11	Selección de dispositivos.....	63
Figura 12	Selección estación San Pedro Ahuacatlán.	64
Figura 13	Variables climatológicas en Ubidots.	64
Figura 14	Grafica Ubidots temperatura.	65
Figura 15	Sección de descarga.	66
Figura 16	Descarga de base datos.	66
Figura 17	Grafica distribución temperatura.....	68
Figura 18	Grafica distribución humedad.....	68
Figura 19	Grafica distribución presión.	68
Figura 20	Grafica comparativa de temperatura entre las estaciones.	70

INDICE DE TABLAS

Tabla 1	Valores Sensor BME680.....	50
Tabla 2	Comparativa temperaturas máximas	67

INDICE DE ILUSTRACIONES

Ilustración 1	Pseudocódigo inicialización de sensores.	52
Ilustración 2	Pseudocódigo recolección de datos y transmisión a la nube.	53
Ilustración 3	Pseudocódigo Limpieza de datos.	57
Ilustración 4	Pseudocódigo análisis climático.....	58
Ilustración 5	Estación meteorológica.....	61
Ilustración 6	Conexión sistema de medición.....	61

INDICE DE ECUACIONES

Ecuación 1	Consumo promedio por ciclo.	62
Ecuación 2	Duración de la batería.....	63

1. INTRODUCCIÓN

El cambio climático está teniendo un gran impacto en el clima, provocando cada vez más fenómenos meteorológicos extremos, intensos y frecuentes. Se espera que el calentamiento global aumente la frecuencia e intensidad de los fenómenos meteorológicos extremos, como tormentas, huracanes, sequías y olas de calor, según un informe del Panel Intergubernamental sobre el Cambio Climático (*Climate Change*, 2021). El cambio climático también afecta la formación y el comportamiento de los sistemas meteorológicos. Por ejemplo, el aumento de la temperatura de los océanos provoca ciclones tropicales más intensos y duraderos, mientras que el derretimiento de los glaciares y los casquetes polares puede afectar la circulación oceánica y atmosférica y cambiar los patrones climáticos a escala mundial (*Climate Change*, 2021).

En su estudio realizado por Peterson et al.(2013) destacó la importancia de utilizar estaciones meteorológicas autónomas para monitorear el clima y el cambio climático. Estas estaciones miden variables como la temperatura, humedad, presión barométrica y velocidad del viento, brindando información valiosa para comprender las condiciones climáticas.

Las estaciones meteorológicas autónomas son una tecnología innovadora que está cambiando la forma en que se recopilan datos meteorológicos. Dichas estaciones recopilan datos meteorológicos precisos en tiempo real sin intervención humana. Además, pueden funcionar con energía renovable para reducir el impacto ambiental y hacer que la recopilación de datos sea más sostenible. Las estaciones meteorológicas autónomas pueden equiparse con sensores para monitorear otros factores ambientales como la calidad del aire y la radiación solar (Morón et al., 2018). Esta capacidad de monitoreo adicional puede proporcionar información valiosa para la toma de decisiones en energías renovables, agricultura y gestión ambiental. Dichas estaciones desempeñan un papel importante en la adaptación al cambio climático al proporcionar datos climáticos precisos y en tiempo real para ayudar en la toma de decisiones en diversas industrias (Math & Dharwadkar, 2018).

La información de las estaciones meteorológicas autónomas es crucial para planificar medidas de adaptación climática. Las medidas de adaptación climática se refieren a las acciones y estrategias para reducir la vulnerabilidad y aumentar la resiliencia de los sistemas y sociedades a los impactos del cambio climático. Las estaciones meteorológicas autónomas pueden proporcionar información valiosa para la planificación e implementación de estas medidas de adaptación. Por ejemplo, los datos recopilados por estaciones meteorológicas autónomas pueden usarse para predecir sequías y determinar la mejor manera de administrar los recursos hídricos en una región determinada (Sepulcre-Canto et al., 2012).

Las estaciones meteorológicas autónomas también son herramientas valiosas para la investigación y la toma de decisiones en una variedad de industrias, desde la agricultura hasta la planificación urbana y la gestión de desastres (Siems-Anderson et al., 2019). La recopilación de datos meteorológicos precisos y en tiempo real proporcionados por estas estaciones puede ayudar a los legisladores y planificadores a tomar decisiones informadas sobre cómo adaptarse al cambio climático y otros factores ambientales. En un mundo cada vez más afectado por el cambio climático y otros factores ambientales, estas estaciones son herramientas importantes para la investigación y la toma de decisiones en diversas industrias como se menciona en Kumar et al., (2021) al utilizar la información climatológica para identificar sequias en la India.

En la actualidad, las estaciones meteorológicas autónomas son una tecnología cada vez más accesible y popular en todo el mundo (Malozyomov et al., 2023). Generalmente las estaciones ofrecen una solución asequible y eficiente para la recolección de datos meteorológicos en áreas remotas y difíciles de acceder, lo que es particularmente útil en regiones donde el cambio climático y otros factores están generando condiciones meteorológicas cada vez más extremas. Además, el avance en la tecnología de la energía renovable ha permitido la implementación de estaciones meteorológicas autónomas que funcionan con energía solar y eólica (Korolkov et al., 2018). A pesar de los avances tecnológicos en el campo de las estaciones meteorológicas, muchas de ellas siguen teniendo limitaciones en cuanto

a su capacidad de almacenamiento, costos y precisión. Por lo tanto, existe la posibilidad de seguir mejorando y aportando soluciones novedosas que permitan una mayor autonomía, reducción de costos y reducción de emisiones de carbono en la recopilación de datos meteorológicos.

2. ANTECEDENTES

A lo largo de la historia, desde los primeros intentos basados en observaciones empíricas hasta los modelos matemáticos y las tecnologías modernas de observación y medición, la meteorología ha evolucionado considerablemente. Se han desarrollado instrumentos de medición precisos, como termómetros y barómetros, para medir la temperatura y la presión atmosférica. En el siglo XIX, se estableció una red mundial de estaciones meteorológicas y se comenzaron a utilizar satélites meteorológicos para obtener datos de la alta atmósfera como menciona Martín, (2018). A lo largo del siglo XX, las técnicas de observación y diagnóstico avanzaron, incluyendo el uso de radares y satélites para obtener imágenes en tiempo real de las condiciones meteorológicas globales, mejorando así la capacidad de predecir el clima.

Las estaciones meteorológicas están compuestas de diferentes sensores como se muestra en la Figura 1. Entre estos sensores se incluyen una veleta para la medición de la dirección del viento, un pluviómetro especializado en la recopilación de datos sobre la precipitación, y un termómetro tipo SIX utilizado para registrar las temperaturas máximas y mínimas diarias. Además, se emplea un tanque evaporimétrico para medir la evaporación efectiva del entorno.

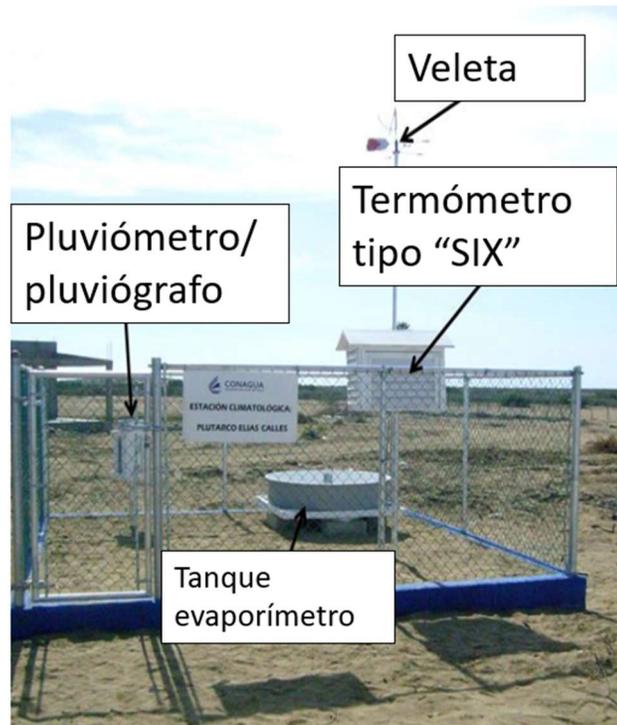


Figura 1 Estación Meteorológica "Plutarco Elías Calles" (CONAGUA, 2005)

Investigaciones respecto a estaciones meteorológicas han sido realizadas en diversas partes del mundo, cada una con objetivos diferentes, por ejemplo, referente a la necesidad de agua y riego mediante el análisis usando estaciones meteorológicas en La Mancha, España realizó las estimaciones de riego mediante la exploración de la precipitación, así como el contenido de agua en el suelo específicamente en viñedos.

La estación meteorológica "Juan Kalt Bode" en la Norpatagonia chilena ha sido una fuente invaluable de registros meteorológicos desde 1935 hasta la actualidad. Los inmigrantes germanos han sido protagonistas en esta historia, y la estación "Juan Kalt Bode" ha sido testigo y proveedor de datos durante más de 70 años. Esta información, ha desempeñado un papel fundamental en el desarrollo de la ciencia meteorológica en la región chilena (Martinez, 2022).

Respecto al uso de unidades de procesamiento, se llevó a cabo en Cataluña, España el diseño e implementación usando como unidad de procesamiento una tarjeta Raspberry Pi para el almacenamiento de los datos climáticos de manera local (Garcia, 2016).

Referente al diseño de estaciones meteorológicas de bajo costo en Turquía Sabanci et al., (2019) plantearon el diseño de una estación meteorológica de bajo costo usando como unidad de procesamiento una tarjeta Arduino Mega 2560 de código abierto, en el que se usaron sensores barométricos así como de humedad y temperatura, sin embargo los datos eran almacenados de manera local por lo que se debían extraer in situ los datos para su análisis o visualización.

Bermeo & Fernando, (2023) en Ecuador, realizaron una investigación donde se resaltaron la importancia de la precisión y calidad de los datos recopilados por estaciones meteorológicas, se propone una red de comunicación inalámbrica para las estaciones meteorológicas en la isla San Cristóbal como principal objetivo el resolver la problemática en la transmisión de datos para el estudio del clima, así como el descenso de la productividad agrícola.

Una parte importante es la validación de los datos meteorológicos así como la aplicación de estos, en este caso como explican Estévez & Gavilán (2008), la validación de datos meteorológicos es una parte importante en la recopilación de información precisa. Este proceso permite identificar registros erróneos y resolver problemas de calibración de sensores. En la Red de Información Agroclimática de Andalucía, se ha implementado este enfoque con éxito, revelando la presencia de registros potencialmente incorrectos. Gracias a un análisis temporal exhaustivo, se han tomado decisiones informadas, como reemplazar sensores defectuosos o depurar registros, para garantizar la fiabilidad de los datos en aplicaciones agronómicas. Este enfoque riguroso asegura que los agricultores y profesionales del sector puedan confiar en la precisión de la información meteorológica para tomar decisiones fundamentadas.

Según el estudio realizado por Virgen-Cobos et al., (2022) en Zoquiapan, Puebla, la localización es una parte esencial del monitoreo, ya que permite estudiar el comportamiento de ambos elementos. Una ubicación adecuada puede proporcionar datos valiosos y precisos para analizar los efectos del cambio climático en la vegetación y su impacto en el medio ambiente.

De manera local en el estado de Querétaro, se han realizado trabajos referentes a estaciones meteorológicas, entre ellos se puede mencionar el trabajo llevado a cabo por Rabiela, (2010) en el cual se buscó el desarrollo y la implementación de una estación meteorológica utilizando radiofrecuencias para la comunicación la cual tuvo lugar en la Universidad Autónoma de Querétaro. Siendo la desventaja principal que los datos se debían descargar de la unidad de procesamiento para su análisis por lo que en la actualidad necesitaría una mejora en el proceso. De la misma forma como la adquisición de las variables climáticas es importante, lo mismo es su análisis. Referente al análisis, este se llevó a cabo en la misma universidad por Muñoz-Mandujano et al., (2022) donde se realizó la implementación de modelos y algoritmos para el pronóstico de precipitaciones extremas aplicado a la red de monitoreo RedCIAQ para el monitoreo y prevención de las condiciones climatológicas.

La revisión de los antecedentes internacionales, nacionales muestra que la incorporación del IoT a las estaciones meteorológicas permite que los datos puedan ser verificables y de rápido acceso. En cuanto al IoT, se refiere a la interconexión de objetos cotidianos a través de Internet. Estos objetos, que van desde dispositivos electrónicos domésticos hasta vehículos y sensores medioambientales, están equipados con sensores, procesadores y conectividad a Internet para comunicarse y compartir datos. Los primeros desarrollos de IoT se remontan a la década de 1990 (Ioannou et al., 2021; Thoutam, 2021), cuando la industria y la logística comenzaron a utilizar sensores y dispositivos de seguimiento para mejorar la eficiencia y el control de los procesos. A medida que la conectividad a Internet se expandió y los costos de los sensores y dispositivos conectados disminuyeron en la década de 2000, el IoT se extendió a sectores como la sanidad, la energía y el transporte (Zeadally et al., 2020). Con el tiempo, se han desarrollado estándares y tecnologías para la interconexión y el procesamiento de datos de objetos conectados. Además, se ha integrado la inteligencia artificial y el aprendizaje automático en el IoT para mejorar la capacidad de análisis y automatización de procesos como por ejemplo realizo Misra et al., (2022) utilizando análisis de datos así como la inteligencia artificial en la agricultura y la industria de la comida.

Las estaciones meteorológicas, los componentes electrónicos y el software con el que operan, están sujetos a normativas, respecto a este tema, las normas son generadas por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE), estas desempeñan un papel esencial en la ingeniería eléctrica, electrónica y de software. Algunas normas relevantes para las pruebas de laboratorio en ingeniería y sistemas embebidos son la IEEE 1012 (*IEEE Standard for System and Software Verification and Validation*, 2012), la IEEE 1220 (Doran, 2006) y la IEEE 829 (documentación de pruebas de software) .

3. DESCRIPCIÓN DEL PROBLEMA

Las estaciones meteorológicas modernas son dispositivos tecnológicos que permiten medir numerosas variables climáticas, como la temperatura, la humedad, la presión atmosférica, la velocidad y dirección del viento y la radiación solar, entre otras. Sin embargo, las estaciones requieren personal especializado para el manejo del software requerido debido a la complejidad de las operaciones y el análisis de datos que se realiza. Estos dispositivos son de elevado costo debido a la sofisticada tecnología y los materiales de alta calidad utilizados en su construcción, así como a la necesidad de mantener y calibrar regularmente los instrumentos para garantizar mediciones precisas y confiables a lo largo del tiempo. Además, el mantenimiento llega a ser costoso por varias razones. En primer lugar, la necesidad de calibrar y verificar regularmente los instrumentos de medición para mantener la precisión puede requerir equipos especializados y personal capacitado, lo que implica costos adicionales. Otro caso es debido a que a menudo se encuentran en ubicaciones remotas o expuestas a condiciones climáticas extremas, lo que aumenta la dificultad y el costo de su mantenimiento. Además, las piezas y componentes de alta calidad utilizados en estas estaciones pueden ser costosos de reemplazar si se dañan o desgastan.

El uso de la infraestructura, así como un sistema embebido especializado al utilizar la tecnología IoT ofrecería mejoras significativas en la recolección de datos y el monitoreo remoto de las estaciones siendo el principal problema en las estaciones modernas, disminuyendo así la dependencia de personal especializado

en el sitio y permitiendo realizar ajustes y calibraciones de manera más eficaz. Además, la integración con el sistema embebido simplificaría la gestión de datos y análisis, lo que haría que mantener estas estaciones en ubicaciones remotas o expuestas a condiciones climáticas extremas resultará más accesible y rentable, así como la facilidad de integración con los diferentes sensores en un solo producto.

4. JUSTIFICACIÓN

El cambio climático global afecta a todos los seres vivos. Las estaciones meteorológicas son vitales para estudiar sus efectos, como el aumento de temperatura y fenómenos extremos. Esto impacta en economía, seguridad alimentaria y salud. Un sistema de monitoreo climático eficiente, sostenible y económico es crucial para la sociedad e industria.

La integración de sistemas embebidos y el IoT en estaciones meteorológicas ofrece ventajas sustanciales en comparación con las soluciones tradicionales. Además de mantener una conexión permanente a Internet, estas tecnologías reducen los costos de implementación y permiten la expansión de redes de sensores en áreas remotas, mejorando la cobertura de datos en lugares previamente inaccesibles. Sin embargo, la interconexión entre sistemas de diferentes fabricantes puede ser complicada debido a las diferencias en la construcción de estos sistemas. Aun así, la estandarización juega un papel clave al permitir que los datos recopilados por sistemas embebidos en estaciones meteorológicas sean transmitidos y comprendidos por dispositivos y plataformas IoT de terceros que sigan los mismos estándares, garantizando la coherencia en la comunicación y la interoperabilidad en la recopilación y uso de datos climáticos. Llevando a cabo la implementación en un sistema embebido con sensores especializados en el ámbito climático permitirá tener un sistema de monitoreo de bajo costo utilizando la tecnología IoT para la parte de recolección de datos.

5. OBJETIVOS GENERALES Y PARTICULARES

OBJETIVO GENERAL

Desarrollar una estación meteorológica de bajo costo para medir y visualizar variables climáticas en tiempo real utilizando la tecnología del internet de las cosas y energía solar.

OBJETIVOS PARTICULARES

- Seleccionar los componentes necesarios para construir una estación meteorológica de bajo costo mediante un análisis de las variables climáticas a monitorear.
- Identificar una unidad de procesamiento adecuada para implementar tecnología IoT en la estación meteorológica mediante la revisión de las especificaciones técnicas de diferentes unidades de procesamiento.
- Reducir el consumo de energía de la estación meteorológica configurando el procesador en modo de bajo consumo para asegurar una operación a largo plazo.
- Elaborar una estación de carga para el microcontrolador mediante una fuente de alimentación compuesta por una batería y paneles solares, para asegurar un suministro de energía en periodos prolongados de tiempo.
- Establecer una conexión Wi-Fi mediante la configuración del microcontrolador para el almacenamiento de variables climáticas en tiempo real en una plataforma en la nube.
- Desarrollar una interfaz de usuario mediante HTML para visualizar datos meteorológicos en tiempo real,
- Validar el funcionamiento de la estación meteorológica de bajo costo mediante la comparativa con bases de datos de estaciones meteorológicas cercanas para analizar la precisión y confiabilidad de los datos recopilados.

6. PLANTEAMIENTO GENERAL

El proyecto se dividirá en tres etapas clave que se presenta en un diagrama de bloques para comprender y llevar a cabo el desarrollo de un sistema de monitoreo y control de una estación a través de Internet. La Figura 2 representa las tres etapas

fundamentales. En primer lugar, se describirá el "Prototipo de Estación", detallando los componentes de hardware y software necesarios para recopilar datos ambientales. A continuación, se abordará la "Conexión, Envío de Datos y Almacenamiento en Internet", explicando cómo se transmitirán de manera segura y confiable estos datos a través de la web. Por último, se mostrará la "Interfaz de Visualización en Internet", ilustrando cómo se presentarán de manera accesible y significativa los datos recopilados a los usuarios en una plataforma en línea.

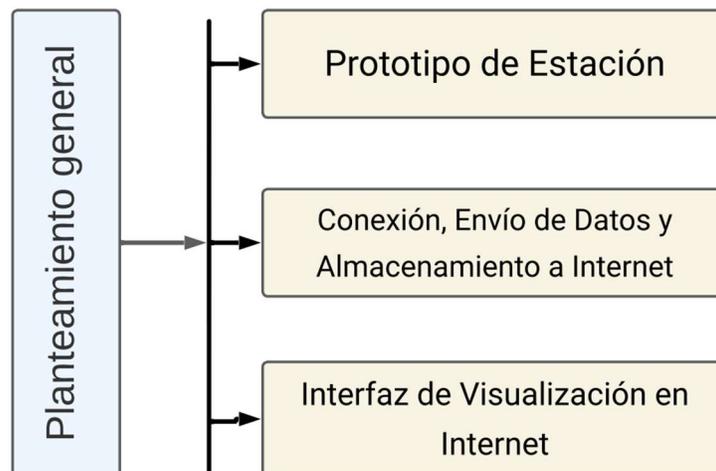


Figura 2 Diagrama de bloques del Planteamiento General.

La sección dedicada al prototipo de la estación implica diversas tareas esenciales, que incluyen la selección y la conexión de sensores, la programación de un microcontrolador para la recopilación de datos, y la realización de pruebas iniciales para asegurar su funcionamiento óptimo.

En lo que respecta a la conectividad, el envío de datos y el almacenamiento, se requiere establecer una conexión a Internet, definir protocolos de comunicación, transmitir los datos desde la estación hasta el servidor en línea y garantizar un almacenamiento seguro de dicha información en la nube.

La sección de Interfaz de visualización en Internet se enfoca en la creación de una interfaz web que permita mostrar de manera accesible y significativa los datos recopilados por la estación de monitoreo en línea. Esto implica tanto el diseño como

el desarrollo, tanto en el lado del cliente como en el servidor, con el fin de presentar y gestionar de manera efectiva los datos recopilados.

7. FUNDAMENTACIÓN TEÓRICA

8.1. ESTACIÓN METEOROLÓGICA

Una estación meteorológica es un dispositivo diseñado para medir y registrar diferentes variables atmosféricas, como la temperatura, la humedad, la presión atmosférica, la velocidad y dirección del viento, la radiación solar y las precipitaciones. Estas estaciones son fundamentales para recopilar datos meteorológicos precisos y confiables, lo que permite a los científicos y meteorólogos comprender mejor los patrones climáticos y predecir el clima futuro (*Climate Change*, 2021).

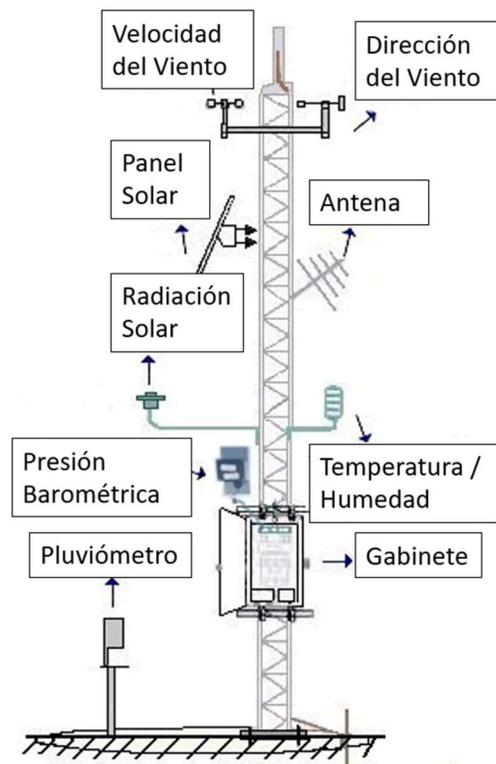


Figura 3 Diagrama de estación meteorológica (Obtenida de CONAGUA).

La Figura 3 ofrece una visión detallada de los componentes esenciales que componen una estación meteorológica. Esta estación se encarga de recopilar datos fundamentales para el análisis del clima y las condiciones atmosféricas. Entre sus elementos destacan:

- **Dispositivos de Medición del Viento:** Esta estación está equipada con instrumentos precisos para medir tanto la dirección como la velocidad del viento llamados veleta y anemómetro. Estos datos son cruciales para comprender los patrones del viento en la ubicación específica de la estación.
- **Sensores de Temperatura, Humedad y Presión:** También se pueden apreciar sensores especializados que monitorean constantemente la temperatura, la humedad relativa y la presión atmosférica. Estos parámetros son esenciales para evaluar las condiciones climáticas y prever cambios significativos en el tiempo.
- **Paneles Solares:** La estación está diseñada de manera sostenible y utiliza energía solar para su funcionamiento. Los paneles solares proporcionan la energía necesaria para alimentar todos los dispositivos y sensores, garantizando así su funcionamiento continuo y respetuoso con el medio ambiente.
- **Pluviómetros:** También se pueden apreciar pluviómetros, que son dispositivos diseñados para medir la cantidad de precipitación, ya sea en forma de lluvia o nieve. Estos datos son cruciales para el seguimiento de la cantidad de precipitación en la región.
- **Gabinetes de Almacenamiento:** En la Figura 3, se observan gabinetes destinados a alojar el equipo de registro y almacenamiento de datos. Estos dispositivos capturan, almacenan y organizan la información recopilada por la estación meteorológica, permitiendo su análisis posterior y su uso en la elaboración de pronósticos y estudios climáticos.

Los datos recopilados por las estaciones meteorológicas se utilizan para realizar análisis climáticos y generar pronósticos meteorológicos precisos. Estos datos también son fundamentales para comprender los cambios climáticos a largo plazo y evaluar su impacto en diferentes áreas, como la agricultura, la aviación y la gestión del agua. Además, las estaciones meteorológicas ayudan a monitorear la calidad del aire y proporcionan información valiosa para la investigación científica y el seguimiento de fenómenos climáticos extremos.

8.2. VARIABLES CLIMÁTICAS

La climatología es la disciplina científica que se encarga de estudiar el clima y sus variaciones a lo largo del tiempo. Una parte fundamental de este estudio es el análisis de las variables climáticas, que son los elementos que permiten caracterizar y cuantificar las condiciones atmosféricas que experimentamos en el entorno (Abbass et al., 2022).

Existen diversas variables climáticas que influyen en el clima de un lugar determinado. Estas variables se pueden clasificar en dos categorías principales: variables básicas y variables derivadas. Las variables básicas son aquellas que se miden directamente y son fundamentales para comprender el clima, mientras que las variables derivadas son aquellas que se obtienen a partir de las mediciones de las variables básicas.

Referente al conjunto de variables climáticas, se distinguen dos categorías fundamentales: las variables básicas y las variables derivadas. Las primeras se revelan como las piedras angulares del entendimiento climatológico, pues se miden de forma directa y proporcionan una perspectiva elemental de las condiciones climáticas. Entre ellas, se cuentan parámetros como la temperatura, la humedad relativa, la presión atmosférica y la velocidad del viento, que son esenciales para esbozar el perfil climático de una región. Por otro lado, las variables derivadas se desprenden de las mediciones de las variables básicas, aportando una visión más elaborada y específica del clima. Ejemplos de estas variables incluyen el índice de calor, la precipitación acumulada, la humedad específica y la dirección del viento, entre otras.

8.3. ESP8266

El microcontrolador ESP8266 ha ganado prominencia como una solución versátil y eficiente en el ámbito de los sistemas embebidos y el Internet de las Cosas (IoT), ofreciendo un abanico de posibilidades para innovadores y desarrolladores. Desde la automatización del hogar y la agricultura inteligente hasta aplicaciones en salud y dispositivos médicos, este microcontrolador se destaca por su capacidad de adaptación a diversas necesidades y entornos. Sin embargo, para aprovechar al

máximo sus capacidades, es esencial comprender a fondo diversos aspectos que abarcan su programación, seguridad, gestión de energía, y potencial para actualizaciones remotas. Cada uno de estos elementos juega un papel crucial en el desarrollo de aplicaciones robustas, seguras y eficientes, permitiendo que el ESP8266 sea una herramienta fundamental en la creación de soluciones tecnológicas avanzadas y sostenibles.

7.3.1. Arquitectura ESP8266

El ESP8266, un sistema en chip (SoC) de gran versatilidad, facilita el desarrollo de sistemas embebidos personalizables con capacidades de conectividad WiFi. Este SoC se caracteriza por una arquitectura integral que permite la implementación de soluciones embebidas avanzadas con funcionalidades de conectividad inalámbrica.

En el núcleo del ESP8266 se encuentra su unidad central de procesamiento, un procesador Tensilica Xtensa L106 de 32 bits, el cual es fundamental en la ejecución de operaciones y procesamiento lógico del sistema (Kodali & Mahesh, 2016). Además, integra una memoria Flash de capacidad variable según el modelo, esencial para el almacenamiento del firmware y el código del usuario.

Un elemento crucial del ESP8266 es su módulo WiFi, que proporciona conectividad inalámbrica a redes existentes en el modo estación, o la capacidad de establecer su propia red en el modo punto de acceso. Respecto a las interfaces periféricas, el ESP8266 ofrece pines de Entrada/Salida de Propósito General (GPIO) para el manejo de señales digitales. De igual manera, dispone de interfaces UART para comunicaciones seriales y soporta SPI e I2C para la interacción con dispositivos periféricos.

Adicionalmente, el ESP8266 está equipado con convertidores analógico-digitales (ADC) para la digitalización de señales analógicas, lo que permite la integración de sensores analógicos. Incluye también temporizadores y contadores para una gestión precisa del tiempo y generación de señales sincronizadas. La arquitectura se complementa con un procesador de señales digitales (DSP) para optimizar operaciones matemáticas y de procesamiento de señales, y un

controlador de energía para una administración eficiente del consumo eléctrico y la implementación de modos de bajo consumo.

Los bloques de entrada/salida (I/O) del ESP8266 son fundamentales para la conexión del chip con otros dispositivos electrónicos, como sensores, actuadores y pantallas, facilitando la interacción y comunicación con el entorno. Estos bloques I/O están estratégicamente organizados para conformar un sistema integrado y funcional.

7.3.2. Diseño de hardware digital

El diseño de hardware digital, un campo crucial en la ingeniería electrónica, implica el desarrollo de circuitos electrónicos que emplean lógica digital para ejecutar operaciones y manejar datos. Este proceso se fundamenta en varios conceptos clave: lógica combinatoria y secuencial, sincronización de señales, y verificación de diseños.

La lógica combinatoria, piedra angular del diseño de hardware digital, se dedica a realizar operaciones lógicas y aritméticas sobre datos en tiempo real. Esta lógica se basa en el uso de compuertas lógicas elementales como AND, OR, NOT, y XOR, formando circuitos que procesan entradas para generar salidas específicas. Los diseñadores utilizan la lógica combinatoria para crear desde circuitos simples, como sumadores y comparadores, hasta complejas arquitecturas encontradas en procesadores y sistemas de control.

Por otro lado, la lógica secuencial se enfoca en operaciones que dependen del estado anterior del sistema. Empleando dispositivos como flip-flops, registros y contadores, que conservan datos y un estado interno, esta lógica permite diseñar circuitos capaces de procesar entradas y generar salidas basadas en condiciones previas y el estado actual. Los circuitos secuenciales son esenciales en aplicaciones donde la memoria y el historial de operaciones juegan un papel fundamental.

La sincronización de señales es otro aspecto crítico, garantizando que las señales en un circuito digital estén temporalmente alineadas. Dada la importancia de la precisión temporal en el procesamiento de datos, las señales asíncronas pueden conducir a errores significativos. Para contrarrestar esto, se emplean

técnicas como el clock gating y el buffering, asegurando la correcta sincronización y el funcionamiento óptimo del circuito.

La verificación de diseños es un proceso meticuloso para confirmar que el circuito digital cumple con las especificaciones y requisitos previstos. Se utilizan métodos de verificación como la simulación, pruebas y análisis formal para validar la integridad del diseño antes de su implementación física. Este paso es crucial para evitar costosos errores y garantizar la fiabilidad y eficacia del hardware en su aplicación final.

7.3.3. Herramientas de diseño

El diseño y programación del ESP8266, un microcontrolador altamente popular para aplicaciones de IoT, se apoya en un conjunto diverso de herramientas de software y hardware. Estas herramientas facilitan desde la descripción del hardware a un nivel de abstracción alto hasta la implementación efectiva del diseño. A continuación, se detallan algunas de estas herramientas clave:

1. **ESP-IDF (IoT Development Framework):** Proveído por Espressif Systems, este framework es el estándar para el desarrollo en el ESP8266. Incluye una amplia gama de bibliotecas y herramientas, abarcando desde componentes de bajo nivel y controladores de hardware hasta ejemplos prácticos, facilitando un control detallado y una optimización eficiente del desarrollo.
2. **GCC Toolchain:** Basado en GNU Compiler Collection, este conjunto de herramientas de compilación cruzada es esencial para generar código ejecutable para el ESP8266. Permite a los desarrolladores escribir código en un entorno familiar y luego compilarlo para la arquitectura específica del ESP8266.
3. **Esptool.py:** Esta herramienta de línea de comandos es clave para la programación y gestión de la memoria flash del ESP8266. Se utiliza para cargar firmware y realizar tareas de mantenimiento, siendo una herramienta indispensable en el flujo de trabajo de desarrollo.

4. **OpenOCD:** Compatible con múltiples adaptadores de depuración JTAG, OpenOCD es una herramienta versátil para la depuración y programación del ESP8266, permitiendo a los desarrolladores depurar programas en tiempo real.
5. **Wireshark:** Conocida por su capacidad para analizar protocolos de red, Wireshark es invaluable para inspeccionar y depurar las comunicaciones de red del ESP8266, ayudando a identificar y solucionar problemas de conectividad.
6. **Visual Studio Code con PlatformIO:** Esta combinación de editor de código y plataforma de desarrollo ofrece un entorno robusto para escribir, compilar y depurar aplicaciones para el ESP8266. Se integra perfectamente con el ecosistema de Espressif, proporcionando una gestión eficiente de proyectos.
7. **Simuladores y Emuladores Personalizados:** Aunque no son comunes debido a la orientación hardware del ESP8266, algunos desarrollos incluyen simuladores o emuladores que permiten evaluar el comportamiento del software antes de su implementación física.
8. **Analizadores Lógicos y Osciloscopios:** Herramientas esenciales para el análisis de hardware, estos dispositivos son cruciales para capturar y analizar señales en el ESP8266, facilitando el diagnóstico en tiempo real y la resolución de problemas de hardware y comunicaciones.
9. **Git y Repositorios de Código:** El uso de sistemas de control de versiones como Git y plataformas de alojamiento como GitHub es fundamental para el desarrollo colaborativo y la gestión eficaz de las versiones del código en proyectos del ESP8266.
10. **Herramientas de Análisis de Consumo Energético:** Considerando la importancia del consumo de energía en sistemas embebidos, diversas herramientas están disponibles para medir y analizar el consumo energético del ESP8266 en diferentes estados operativos, lo cual es crucial para optimizar la eficiencia energética en aplicaciones IoT.

Cada una de estas herramientas desempeña un papel significativo en el ciclo de vida del desarrollo del ESP8266, desde la conceptualización hasta la implementación y el análisis post-despliegue, asegurando que los proyectos sean tanto técnicamente sólidos como eficientes y confiables.

7.3.4. Seguridad en Aplicaciones IoT

En el contexto de las aplicaciones IoT, la seguridad es una preocupación primordial, especialmente cuando se trata de dispositivos como el ESP8266 que se conectan a redes y transmiten datos sensibles. A continuación, se detallan los aspectos clave de la seguridad y el cifrado en el ESP8266, así como su gestión de la seguridad en redes WiFi.

Mecanismos de Seguridad y Cifrado:

Protocolos de Cifrado: El ESP8266 soporta varios protocolos de cifrado estándar para proteger la comunicación de datos. Estos incluyen TLS/SSL (Transport Layer Security/Secure Sockets Layer) para cifrar las comunicaciones a través de Internet y proteger contra interceptaciones y accesos no autorizados.

Autenticación y Autorización: Para asegurar que solo los usuarios y dispositivos autorizados puedan acceder y comunicarse con el ESP8266, se pueden implementar mecanismos de autenticación como contraseñas, tokens de seguridad y claves de acceso.

Integridad de Datos: Se pueden utilizar algoritmos de hash como SHA (Secure Hash Algorithm) para garantizar la integridad de los datos transmitidos, asegurando que no hayan sido alterados durante la transmisión.

Cifrado de Datos en Reposo: Para los datos almacenados localmente en el ESP8266, es aconsejable utilizar técnicas de cifrado para proteger la información sensible, especialmente en aplicaciones que manejan datos personales o críticos.

Gestión de la Seguridad en Redes WiFi:

Autenticación WiFi: El ESP8266 permite la implementación de estándares de seguridad WiFi como WPA/WPA2 (Wi-Fi Protected Access) para autenticar

dispositivos en la red. Esto previene el acceso no autorizado y asegura que solo los dispositivos con las credenciales correctas puedan conectarse a la red.

Acceso Seguro a Redes: Se pueden establecer configuraciones de seguridad adicionales, como la configuración de listas de control de acceso (ACLs) y la ocultación del SSID (Service Set Identifier) para mejorar la seguridad de la red WiFi a la que se conecta el ESP8266.

Seguridad en la Capa de Red: Implementar firewalls y técnicas de segmentación de red para aislar el ESP8266 y otros dispositivos IoT en una red separada, reduciendo el riesgo de ataques cibernéticos a través de la red principal.

Actualizaciones de Seguridad: Mantener el firmware del ESP8266 actualizado es vital para la seguridad. Las actualizaciones OTA (Over-The-Air) facilitan la implementación de parches de seguridad y actualizaciones de software para abordar vulnerabilidades conocidas.

7.3.5. Implementación de algoritmos

La implementación de algoritmos en el ESP8266, un microcontrolador con capacidades de conectividad WiFi, representa un avance significativo en el campo de los sistemas embebidos y el Internet de las Cosas (IoT). La flexibilidad y las amplias opciones de programación que ofrece este dispositivo son fundamentales para su adaptabilidad a una variedad de aplicaciones.

Programación en C/C++: El ESP8266 es comúnmente programado en C/C++, lenguajes que ofrecen un equilibrio entre control a bajo nivel y eficiencia en la ejecución. Esto permite a los desarrolladores implementar algoritmos complejos y optimizados, aprovechando al máximo las capacidades del hardware.

Comunicación Inalámbrica Integrada: Uno de los aspectos más destacados del ESP8266 es su módulo WiFi incorporado, que facilita la implementación de algoritmos que requieren conectividad inalámbrica. Esto lo hace ideal para aplicaciones IoT, donde la transmisión y recepción de datos a través de redes inalámbricas es esencial.

Aplicaciones en Monitoreo en Tiempo Real: La capacidad del ESP8266 para ejecutar algoritmos complejos y comunicarse inalámbricamente lo hace ideal para aplicaciones de monitoreo en tiempo real. Por ejemplo, puede ser utilizado en sistemas de monitoreo ambiental, donde los sensores recolectan datos sobre condiciones como temperatura, humedad o calidad del aire, y el ESP8266 procesa y transmite estos datos para análisis y respuestas en tiempo real.

Automatización de Tareas en Dispositivos Conectados: En el ámbito de la automatización del hogar y la industria, el ESP8266 puede ser programado para controlar y automatizar dispositivos conectados. Desde el control de la iluminación y la temperatura hasta sistemas de seguridad y gestión de energía, la implementación de algoritmos en este microcontrolador permite una gestión eficiente y automatizada de tareas.

Personalización y Adaptabilidad: La verdadera fortaleza del ESP8266 reside en su capacidad para ser programado y adaptado a una amplia gama de aplicaciones específicas. Los desarrolladores pueden crear algoritmos personalizados que se ajusten a las necesidades exactas de su proyecto, ya sea para aplicaciones domésticas, comerciales o industriales.

Desarrollo de Aplicaciones IoT Avanzadas: Con su capacidad para ejecutar algoritmos sofisticados y su módulo WiFi integrado, el ESP8266 es una plataforma ideal para desarrollar aplicaciones IoT avanzadas. Esto incluye desde soluciones de telemetría hasta redes de sensores interconectados y sistemas de control remoto.

7.3.6. Comunicación con periféricos

La comunicación con periféricos es un aspecto fundamental en el diseño de sistemas digitales utilizando el ESP8266, un microcontrolador versátil con amplias capacidades de conectividad. Esta conexión permite integrar una variedad de dispositivos de entrada y salida, como sensores, teclados, pantallas y motores, enriqueciendo las posibilidades de aplicación de este microcontrolador.

Buses de Comunicación: El ESP8266 admite varios buses de comunicación para la conexión de periféricos. Los más destacados son:

I2C (Inter-Integrated Circuit): Un bus de comunicación utilizado para conectar múltiples dispositivos de baja velocidad como sensores y otros componentes. El ESP8266 puede operar como un maestro en la red I2C, facilitando la interacción con diversos dispositivos periféricos.

SPI (Serial Peripheral Interface): Este bus de comunicación es esencial para interactuar con dispositivos que requieren una comunicación de alta velocidad, como ciertas pantallas y sensores. El ESP8266 utiliza el protocolo SPI para una comunicación eficaz y rápida con estos periféricos.

UART (Universal Asynchronous Receiver/Transmitter): Este protocolo de comunicación serie es fundamental para la transmisión de datos entre dispositivos. La UART en el ESP8266 se basa en una transmisión asíncrona, utilizando líneas de transmisión y recepción separadas.

Conectividad WiFi (802.11 b/g/n): Una de las características más notables del ESP8266 es su módulo WiFi incorporado, que posibilita la conexión a redes inalámbricas y la comunicación con otros dispositivos a través de redes locales o Internet. Este microcontrolador puede funcionar tanto como cliente en redes WiFi-existentes como punto de acceso WiFi propio.

Protocolos de Red:

TCP/IP: Utilizando los protocolos TCP e IP, el ESP8266 maneja la comunicación en redes IP, lo que es crucial para la transmisión de datos a través de Internet y para establecer conexiones confiables.

UDP (User Datagram Protocol): Este protocolo, menos complejo que TCP, permite la transmisión rápida de paquetes de datos sin necesidad de una conexión previa, ideal para aplicaciones que requieren rapidez, aunque pueden tolerar cierta pérdida de datos.

MQTT (Message Queuing Telemetry Transport): Un protocolo de mensajería ligero y eficiente, ideal para dispositivos con limitaciones de ancho de banda y recursos, ampliamente utilizado en IoT para la comunicación entre dispositivos y servidores.

HTTP (Hypertext Transfer Protocol): Aunque comúnmente asociado con la navegación web, el ESP8266 puede implementar servidores HTTP para interfaces de usuario basadas en navegadores y comunicación con servicios en línea.

WebSocket: Proporciona una comunicación bidireccional y de baja latencia entre un servidor y un cliente, ideal para aplicaciones en tiempo real.

ESP8266 en sistemas embebidos

El ESP8266 se ha consolidado como una plataforma líder en el ámbito de los sistemas embebidos y el Internet de las Cosas (IoT), en gran medida gracias a sus avanzadas capacidades de comunicación WiFi. Esta funcionalidad integral no solo ha redefinido la interacción y transmisión de datos entre dispositivos interconectados, sino que también ha ampliado el alcance de sus aplicaciones en diversos sectores.

7.3.7. Gestión de Energía del ESP8266: Optimización Eficiente

En el diseño de sistemas embebidos y aplicaciones IoT, la gestión eficiente de la energía es crucial, y el ESP8266 sobresale en este aspecto. La optimización energética en el ESP8266 no solo extiende la vida útil de la batería en dispositivos portátiles, sino que también minimiza los costos operativos y fomenta prácticas sostenibles. Las estrategias y técnicas de gestión de energía en el ESP8266 incluyen:

1. **Modos de Suspensión e Hibernación:** Estos modos reducen o apagan el consumo de energía de componentes no esenciales cuando están inactivos, prolongando la duración de la batería.
2. **Wake-Up Externo:** Permite que el ESP8266 se active en respuesta a señales o interrupciones externas, manteniendo el dispositivo en bajo consumo hasta que sea necesario.
3. **Administración de Periféricos:** La capacidad de activar o desactivar periféricos según sea necesario optimiza el uso de energía, especialmente en reposo.

4. **Control de Velocidad de Reloj:** Ajustar la frecuencia del procesador puede reducir el consumo de energía, útil en aplicaciones donde el rendimiento máximo no es siempre necesario.
5. **Uso Eficiente de WiFi:** Dado que la conectividad WiFi es uno de los principales consumidores de energía, su gestión eficiente es clave para optimizar la autonomía del dispositivo.
6. **Explotación de Energías Renovables:** La integración con fuentes de energía renovable como paneles solares puede aumentar la autonomía en aplicaciones IoT remotas.

Actualización Remota a Través del Aire (OTA) con ESP8266: Flexibilidad y Eficiencia en Sistemas Embebidos

La funcionalidad de Actualización Remota a Través del Aire (OTA) en el ESP8266 ha revolucionado el mantenimiento y la evolución de sistemas embebidos e IoT. Esta característica permite la actualización de firmware y software sin necesidad de conexiones físicas, ofreciendo numerosas ventajas:

7. **Actualizaciones Sin Cables:** La OTA elimina la necesidad de acceso físico a los dispositivos, lo que es crucial en aplicaciones distribuidas o remotas.
8. **Agilidad en el Desarrollo y Mantenimiento:** Facilita la implementación rápida de mejoras y correcciones, beneficiando tanto a desarrolladores como a usuarios finales.
9. **Minimización del Downtime:** Las actualizaciones en segundo plano permiten que los dispositivos sigan operando durante el proceso, crucial en aplicaciones críticas.
10. **Adaptabilidad:** Permite que los dispositivos se adapten a cambios en requisitos o entornos, esencial en el dinámico mundo del IoT.
11. **Escalabilidad:** Facilita la actualización de múltiples dispositivos simultáneamente, simplificando la gestión en proyectos de gran escala.

12. **Reducción de Costos:** Disminuye los costos operativos y de mantenimiento al evitar actualizaciones manuales o envíos de dispositivos.
13. **Seguridad en Actualizaciones:** Soporta protocolos de seguridad para asegurar que solo se apliquen actualizaciones autorizadas.
14. **Mejora Continua:** Permite una evolución constante de los sistemas, prolongando su vida útil y valor.
15. **Interfaz de Usuario Simplificada:** Mejora la experiencia del usuario al simplificar el proceso de actualización.

7.3.8. ESP8266 en aplicaciones específicas

El microcontrolador ESP8266 ha emergido como una plataforma destacada en el ámbito de los sistemas embebidos y el Internet de las Cosas (IoT), gracias a su capacidad para ejecutar cálculos altamente personalizados. Esta versatilidad lo hace ideal para una amplia gama de aplicaciones especializadas, algunas de las cuales se detallan a continuación:

Automatización del Hogar (Domótica): El ESP8266 está revolucionando la interacción con el hogar, ofreciendo mayor comodidad, eficiencia energética, seguridad y personalización. Su uso como componente central en la domótica permite implementaciones sofisticadas con una inversión razonable y gran flexibilidad:

- **Control Remoto y Automatización:** Facilita el manejo de dispositivos domésticos como luces, enchufes, termostatos y cortinas, permitiendo la programación y operación remota.
- **Comunicación Inalámbrica:** La conectividad WiFi del ESP8266 simplifica la integración de dispositivos en la red doméstica y posibilita el control a través de dispositivos móviles y computadoras.
- **Sensores y Monitoreo:** Permite el uso de sensores para recopilar datos ambientales y de presencia, facilitando la toma de decisiones automáticas y la adaptación del ambiente doméstico a las necesidades de los usuarios.

Agricultura y Agricultura Inteligente: El ESP8266 es una herramienta valiosa para optimizar las operaciones agrícolas mediante la recopilación y análisis de datos en tiempo real, lo que resulta en una gestión más eficiente y sostenible:

- **Monitoreo de Condiciones del Suelo:** Sensores conectados al ESP8266 pueden medir variables críticas como la humedad del suelo y los niveles de nutrientes, optimizando el riego y la fertilización.
- **Automatización de Riego:** Se emplea para controlar sistemas de riego automatizados, ajustando la irrigación según las condiciones del suelo y climáticas.
- **Monitoreo Climático:** Sensores meteorológicos brindan información en tiempo real, permitiendo anticipar y mitigar impactos climáticos en los cultivos.
- **Uso Eficiente de Recursos:** Optimiza el consumo de agua y fertilizantes, contribuyendo a la sostenibilidad y la reducción de costos.

Salud y Dispositivos Médicos: El ESP8266 ha demostrado ser una herramienta poderosa en el desarrollo de dispositivos médicos inteligentes y soluciones innovadoras de salud:

- **Monitoreo de Signos Vitales:** Interactúa con sensores biométricos para medir y transmitir datos vitales como la frecuencia cardíaca y la presión arterial.
- **Dispositivos de Monitoreo Continuo:** Rastrea parámetros como la actividad física y el sueño, proporcionando datos valiosos para la evaluación del bienestar.
- **Alertas y Notificaciones:** Capaz de enviar alertas en tiempo real a dispositivos móviles o profesionales médicos ante condiciones anormales.
- **Gestión de Enfermedades Crónicas:** Permite a los pacientes con enfermedades crónicas monitorear su estado y comunicar datos importantes a sus médicos para ajustar tratamientos.

- **Telemedicina y Consultas a Distancia:** Facilita consultas virtuales y seguimiento a distancia, mejorando el acceso a la atención médica en áreas remotas o situaciones de emergencia.

8.4. Protocolos de comunicación

Los protocolos de comunicación, en este contexto, actúan como los pilares fundamentales que permiten la interacción eficiente y efectiva entre el ESP8266 y los dispositivos periféricos. Cada protocolo posee características únicas y cumple un rol específico en el ecosistema IoT, asegurando que la transferencia de datos sea fluida, segura y optimizada para las necesidades de cada aplicación. En particular, cuando se utiliza el ESP8266 en conjunto con sensores SPI e I2C, se activan distintos mecanismos de comunicación, cada uno con sus propias ventajas, limitaciones y aplicaciones ideales.

Esta sección profundiza en los diversos protocolos de comunicación que se emplean en conjunción con el ESP8266, detallando sus características, aplicaciones y la sinergia que crean dentro de proyectos IoT. Se examinarán los protocolos SPI (Serial Peripheral Interface) e I2C (Inter-Integrated Circuit), destacando cómo estos facilitan la comunicación entre el ESP8266 y una variedad de sensores y dispositivos. Además, se abordará el papel de las entradas analógicas y la conectividad a internet, elementos clave para la expansión y la funcionalidad de los sistemas basados en el ESP8266.

7.4.1. SPI

El protocolo SPI es un método de comunicación serial síncrono ampliamente utilizado para la interconexión de dispositivos electrónicos. Desarrollado originalmente por Motorola, SPI facilita una comunicación rápida y eficiente entre un microcontrolador principal (como el ESP8266) y uno o más dispositivos periféricos. Se utiliza comúnmente para conectar microcontroladores a diversos dispositivos periféricos como sensores, memorias, y otros módulos de interfaz.

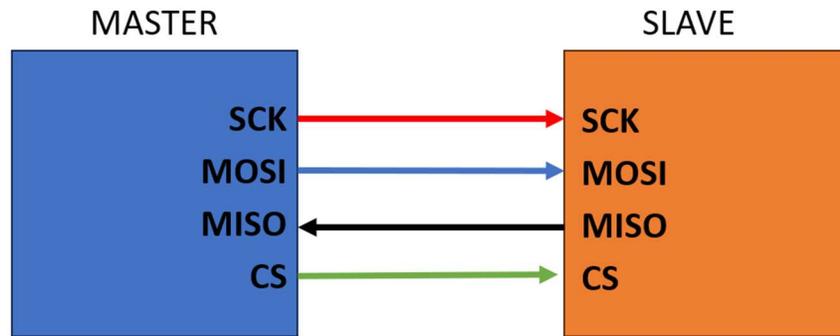


Figura 4 Conexión SPI (Autoría propia).

En la Figura 4, se ilustra la conexión estándar del protocolo SPI, un sistema de comunicación serial síncrona que juega un papel vital en la interconexión de dispositivos en proyectos IoT. Este esquema específico resalta la topología punto a punto entre un dispositivo maestro y uno esclavo, que es fundamental en el funcionamiento del módulo ESP8266 cuando opera en modo maestro.

Las líneas de conexión representadas en distintos colores corresponden a las siguientes señales fundamentales del protocolo SPI:

- **SCK (Serial Clock):** Indicada por la línea roja, esta señal de reloj es esencial para la sincronización de la comunicación serial entre el maestro y el esclavo. El dispositivo maestro genera esta señal para orquestar el intercambio de datos con precisión temporal.
- **MOSI (Master Output, Slave Input):** Mostrada con una línea azul, esta vía se utiliza para transmitir los datos desde el maestro al esclavo. Permite el envío de comandos o datos que el dispositivo esclavo requiere para su operación o configuración.
- **MISO (Master Input, Slave Output):** La línea negra señala la ruta por la cual los datos son enviados de vuelta al maestro desde el dispositivo esclavo. Esta retroalimentación es crítica para la adquisición de datos y el monitoreo de los dispositivos periféricos.
- **CS (Chip Select):** La señal de selección de chip, ilustrada con una línea verde, es utilizada por el maestro para activar un esclavo específico en un sistema donde múltiples dispositivos están presentes en la red SPI. Esta

línea permite al maestro seleccionar con quién comunicarse en un momento dado.

7.4.2. I2C

El protocolo I2C (Inter-Integrated Circuit), también conocido como TWI (Two Wire Interface), es un protocolo de comunicación serial que se ha establecido como un estándar de facto para la comunicación entre circuitos integrados. La simplicidad de I2C reside en su uso de solo dos líneas bidireccionales:

- **SDA (Serial Data):** Esta línea es responsable del intercambio de datos entre dispositivos. Permite tanto la transmisión como la recepción de datos entre el maestro y los dispositivos esclavos conectados.
- **SCL (Serial Clock):** Operando en conjunto con la línea SDA, esta línea de reloj es utilizada para sincronizar la transferencia de datos entre dispositivos, garantizando la integridad y la coherencia de la información intercambiada.

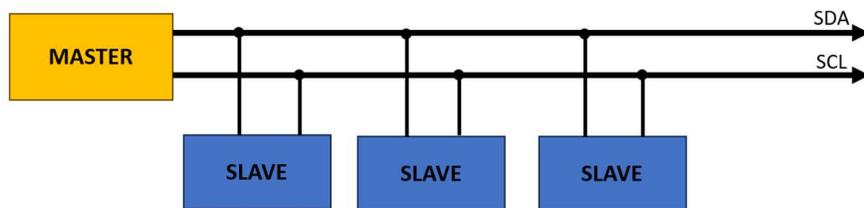


Figura 5 Conexión I2C (Autoría propia).

Una de las ventajas más significativas de I2C sobre otros protocolos de comunicación es su capacidad para conectar múltiples dispositivos esclavos al microcontrolador ESP8266 utilizando solo dos pines, SDA y SCL como se muestra en la Figura 5. Esto no solo ahorra valiosos pines de E/S en proyectos de IoT que a menudo se ven limitados por el espacio físico y los recursos de hardware, sino que también simplifica el diseño del circuito al reducir la cantidad de rutas de cableado necesarias.

Este protocolo es apreciado por su flexibilidad y eficiencia, ya que puede operar en varios modos de velocidad, como el estándar (100 Kbps), el rápido (400 Kbps) y el de alta velocidad (3.4 Mbps), lo que lo hace adaptable a las necesidades

específicas del sistema de comunicación. Además, su naturaleza de bus abierto facilita la expansión del sistema sin necesidad de reconfigurar cada dispositivo individualmente.

7.4.3. GPIO (General Purpose Input/Output)

Los pines GPIO son interfaces programables que se encuentran en microcontroladores y otros circuitos integrados. En el ESP8266, los pines GPIO son fundamentales para la realización de tareas múltiples, desde la simple lectura de datos de un sensor hasta el control de actuadores complejos.

Estos pines pueden ser configurados en el software para actuar como entradas, donde leen señales externas, o como salidas, donde envían señales a otros dispositivos o circuitos.

- **Entrada Digital:** En este modo, los pines GPIO detectan el estado de señales digitales, típicamente '0' o '1', lo cual representa niveles de voltaje bajo o alto respectivamente. Este modo es fundamental para leer datos de sensores digitales o interruptores.
- **Salida Digital:** Cuando se configuran como salidas, los pines GPIO pueden enviar señales digitales, estableciendo un voltaje alto o bajo, lo que es útil para activar relés, encender LEDs o controlar motores.
- **Entrada Analógica:** A diferencia del modo digital, la entrada analógica permite leer un espectro continuo de valores de voltaje, lo cual es esencial para sensores que miden variables continuas como la luz o la temperatura.
- **Salida Analógica:** El modo de salida analógica (a través de modulación por ancho de pulso - PWM) emula una señal analógica permitiendo controlar la intensidad de un LED o la velocidad de un motor.

Es importante señalar que no todos los pines GPIO del ESP8266 soportan todas estas configuraciones. Algunos pines están reservados para funciones específicas o tienen limitaciones en su uso.

7.4.4. ADC (Analog to Digital Converter)

El Convertidor Analógico a Digital (ADC) permite la interfaz entre el mundo analógico y el digital. El ADC transforma señales analógicas, que varían continuamente, en valores digitales que el microcontrolador puede procesar. Esta conversión es esencial para aplicaciones que requieren la monitorización de fenómenos físicos representados a través de señales analógicas.

El ESP8266 está equipado con un único canal ADC que puede medir voltajes de 0 a 1 V. Este rango puede ser extendido hasta el voltaje de alimentación del microcontrolador, normalmente 3.3 V, utilizando circuitos externos como divisores de voltaje. La resolución del ADC en el ESP8266 es de 10 bits, lo que significa que el voltaje medido se representa como un valor digital en un rango de 0 a 1023.

El ADC es especialmente útil para leer datos de una variedad de sensores analógicos. Por ejemplo:

- **Sensores de Temperatura:** A través de termistores o dispositivos similares, se pueden obtener lecturas precisas de temperatura que son convertidas en valores digitales para su procesamiento y análisis.
- **Sensores de Luz:** Foceldas o fotodiodos generan un voltaje proporcional a la intensidad de la luz, que el ADC puede leer y convertir, permitiendo al microcontrolador responder a cambios en la iluminación.
- **Sensores de Humedad:** Los sensores de humedad resistivos varían su resistencia en función de la humedad ambiental, lo que se traduce en un cambio de voltaje que el ADC puede cuantificar.

7.4.5. Protocolos de Internet

El Internet de las Cosas (IoT) se basa en una serie de protocolos de comunicación estandarizados para facilitar la transferencia de datos entre dispositivos y sistemas. El ESP8266 es compatible con varios de estos protocolos esenciales, cada uno con sus propias características y aplicaciones específicas:

HTTP/HTTPS para la Comunicación Web:

HTTP (Hypertext Transfer Protocol) es el protocolo fundamental utilizado para la transmisión de datos en la World Wide Web. En dispositivos IoT como el ESP8266, HTTP permite enviar y recibir datos a través de solicitudes y respuestas estándar de la web, facilitando la integración con aplicaciones y servicios basados en internet.

HTTPS (HTTP Secure) añade una capa de seguridad a las comunicaciones HTTP mediante el uso de cifrado SSL/TLS. Esto asegura que los datos transmitidos entre el ESP8266 y los servidores web sean confidenciales y seguros, lo cual es crítico en aplicaciones donde se manejan datos sensibles o personales.

MQTT para la Mensajería en Proyectos de IoT:

MQTT (Message Queuing Telemetry Transport) es un protocolo de mensajería ligero y eficiente diseñado para entornos con ancho de banda limitado y requiere baja latencia y cargas de red mínimas. Es ideal para la comunicación entre el ESP8266 y un broker MQTT en la nube, permitiendo una gestión eficaz de la mensajería entre dispositivos y aplicaciones IoT. Su modelo de publicación/suscripción facilita la actualización de estados de dispositivos y la telemetría en tiempo real.

WebSockets para una Comunicación Bidireccional en Tiempo Real:

WebSockets proporcionan un canal de comunicación bidireccional y persistente sobre una única conexión TCP. Con WebSockets, el ESP8266 puede participar en sesiones de comunicación en tiempo real, lo que es especialmente valioso para aplicaciones como chatbots, actualizaciones en vivo de sensores y control interactivo de dispositivos IoT. A diferencia de HTTP, donde cada transferencia de datos requiere una nueva conexión, WebSockets mantienen la conexión abierta, lo que reduce la latencia y el overhead.

8.5. SENSORES

Los sensores en las estaciones meteorológicas son componentes esenciales para el monitoreo y análisis del clima. Cada sensor está especializado en medir una

variable ambiental particular, y su precisión es fundamental para obtener datos fiables. (Lin et al., 2021).



Figura 6 Sensor de temperatura BME680.(autoría propia).

Existen diversos tipos de sensores utilizados en las estaciones meteorológicas, cada uno diseñado para medir una variable específica. Por ejemplo, los termómetros se utilizan para medir la temperatura ambiente, mientras que los higrómetros miden la humedad relativa del aire (Bourouina, 2021). Los barómetros miden la presión atmosférica y los anemómetros registran la velocidad y dirección del viento.

La importancia de estos sensores radica en su capacidad para proporcionar datos precisos y en tiempo real sobre las condiciones meteorológicas. Estos datos se utilizan para la predicción del clima, la investigación científica, la planificación de actividades al aire libre, la gestión de desastres naturales y la toma de decisiones en diversas industrias, como la agricultura, la aviación y la energía renovable. A continuación, se describen los tipos de sensores más comunes y su funcionamiento:

Sensor de Temperatura (por ejemplo, BME680): Mide la temperatura del aire. Los sensores como el de la Figura 6 utilizan elementos sensibles al calor para detectar cambios en la temperatura ambiente. Este sensor en particular también puede medir la calidad del aire, la presión atmosférica y la humedad, lo que lo hace muy versátil para aplicaciones meteorológicas.

Higrómetro: Mide la humedad relativa del aire. Los higrómetros modernos suelen utilizar tecnología de condensadores o resistencias que cambian sus propiedades eléctricas en respuesta a variaciones en la humedad del aire.

Barómetro: Registra la presión atmosférica. Los barómetros digitales utilizan sensores de presión que convierten la fuerza ejercida por el aire en una señal eléctrica, lo que permite medir con precisión los cambios en la presión atmosférica.

Anemómetro: Mide la velocidad y dirección del viento. Los anemómetros tradicionales utilizan copas o paletas que giran con el viento; la velocidad de rotación se convierte en una medida de la velocidad del viento. Los anemómetros modernos pueden usar tecnología ultrasónica, donde la velocidad del viento se determina midiendo el tiempo que tarda el sonido en viajar entre dos puntos.

Pluviómetro: Registra la cantidad de precipitación. Funciona recogiendo el agua de lluvia en un contenedor medido. Los pluviómetros más avanzados utilizan sensores para detectar el peso del agua recogida, lo que permite una medición precisa de la precipitación.

Sensor de Radiación Solar: Mide la intensidad de la radiación solar. Estos sensores, a menudo llamados piranómetros, utilizan una superficie que absorbe la radiación solar, convirtiendo el calor en una señal eléctrica que se mide para determinar la intensidad de la luz solar.

Veleta: Determina la dirección del viento. Las veletas tienen una parte móvil que se alinea con la dirección del viento. La orientación de esta parte se traduce en una señal eléctrica que indica la dirección desde la que sopla el viento.

8.6. INTERNET DE LAS COSAS

La tecnología del Internet de las cosas ha revolucionado la forma en que se recopilan datos meteorológicos, se monitorea y se predice el clima. La utilización de sensores y dispositivos conectados a Internet permite recopilar datos en tiempo real y con alta precisión. Esto significa que se puede obtener información meteorológica más detallada y precisa que antes, lo que puede tener importantes aplicaciones en la agricultura, la aviación, la navegación y otras áreas. Además, la tecnología del IoT también puede ayudar a predecir el clima con mayor precisión, lo que es vital en situaciones de emergencia, como desastres naturales(Li et al., 2020).

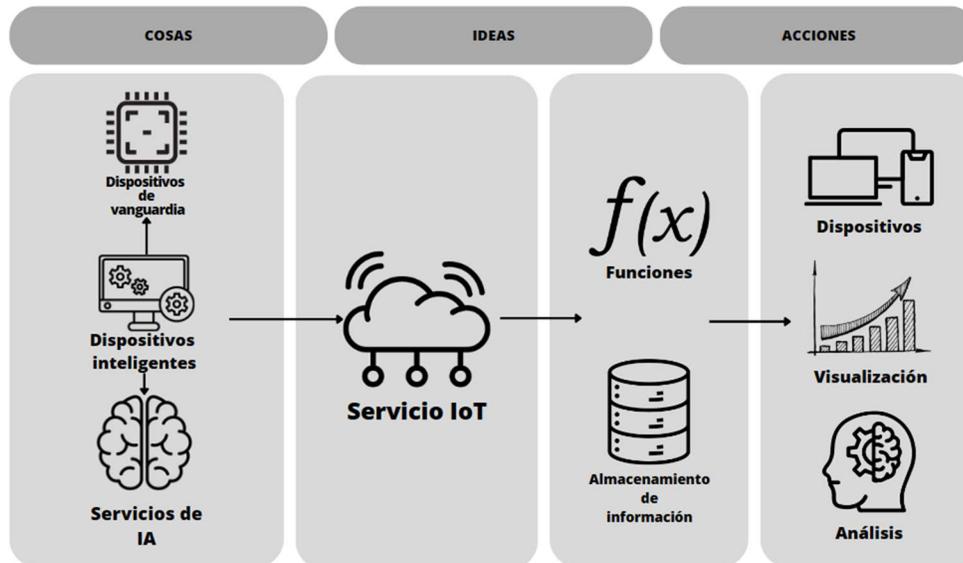


Figura 7 Diagrama lot (Autoría propia).

La Figura 7 representa el diagrama principal de un ecosistema IoT (Internet de las cosas), en el cual los dispositivos de vanguardia desempeñan un papel fundamental en la recopilación de información. Además, se emplean servicios de inteligencia artificial (IA) para el control y gestión de esta información. Los dispositivos inteligentes nos permiten establecer una conexión con el servicio IoT, que en la mayoría de los casos se encuentra alojado en un servicio de almacenamiento en la nube. Todo esto tiene como finalidad enriquecer el repositorio de datos con las variables necesarias para llevar a cabo un análisis de datos mediante diversas funciones.

Una vez realizado el respectivo análisis, el potencial del IoT radica en la posibilidad de acceder a esta información desde cualquier lugar del mundo, utilizando un dispositivo inteligente para su visualización y aprovechamiento.

7.5.1. Análisis y procesamiento de datos

Esta sección profundiza en cómo los datos recolectados son analizados y procesados para obtener información significativa y accionable. Abordaremos las técnicas y herramientas utilizadas para filtrar, analizar y sintetizar grandes conjuntos de datos, transformándolos en conocimientos útiles para la toma de decisiones. Desde el procesamiento en tiempo real hasta el análisis predictivo, exploraremos

cómo el tratamiento inteligente de los datos puede mejorar la eficiencia operativa, la precisión en las predicciones meteorológicas y el impacto en diversas aplicaciones prácticas.

Estas etapas incluyen:

Recolección de Datos:

- Esta etapa implica la recopilación de datos brutos de diversos sensores. En una estación meteorológica, esto puede incluir datos de temperatura, humedad, presión atmosférica, velocidad del viento, y más. Es crucial que la recolección de datos sea precisa y confiable para asegurar la calidad del análisis posterior.

Filtrado y Preprocesamiento:

- Antes del análisis, los datos pueden requerir limpieza y filtrado. Esto incluye eliminar datos erróneos o atípicos, corregir errores, y normalizar los datos para análisis. El preprocesamiento también puede implicar la conversión de datos a formatos adecuados para el análisis.

Análisis de Datos:

- En esta fase, se aplican técnicas estadísticas y de aprendizaje automático para interpretar los datos. Esto puede variar desde análisis simples, como calcular promedios y tendencias, hasta modelos predictivos más complejos que pueden pronosticar condiciones futuras basadas en datos históricos.

Visualización de Datos:

- Los datos analizados a menudo se presentan mediante visualizaciones como gráficos, mapas de calor y dashboards. Estas herramientas ayudan a los usuarios a comprender fácilmente patrones complejos y tendencias en los datos.

Almacenamiento de Datos Post-Análisis:

- Los datos procesados y las conclusiones derivadas de ellos necesitan almacenarse para su uso futuro. Esto puede implicar bases de datos que soporten consultas eficientes y acceso a datos históricos.

Toma de Decisiones Basada en Datos:

- Finalmente, los resultados obtenidos del análisis de datos se utilizan para tomar decisiones informadas. En el contexto de una estación meteorológica, esto podría influir en la planificación agrícola, la gestión de recursos naturales, o las alertas de condiciones climáticas adversas.

Retroalimentación y Mejora Continua:

- Basándose en el análisis y la aplicación de los datos, el sistema puede ser continuamente mejorado. Por ejemplo, ajustes en los algoritmos de recopilación y análisis de datos pueden ser realizados para aumentar la precisión y la eficiencia del sistema.

7.5.2. Almacenamiento de datos

En el dinámico y creciente campo del Internet de las Cosas (IoT), el Almacenamiento de Datos se presenta como un componente fundamental que impacta significativamente en la eficacia y eficiencia de cualquier sistema IoT, incluyendo estaciones meteorológicas avanzadas. Esta sección explora los aspectos técnicos clave del almacenamiento de datos, abarcando desde la evaluación de la capacidad y escalabilidad necesarias para manejar volúmenes crecientes de información, hasta la integridad y fiabilidad de los datos almacenados. También se considera la importancia de la accesibilidad y la recuperación de datos, así como las diferencias críticas y consideraciones entre el almacenamiento en la nube y el almacenamiento local. Estos factores son esenciales para garantizar que los sistemas IoT, como las estaciones meteorológicas que utilizan tecnologías como el ESP8266, no solo recolecten datos de manera eficiente, sino que también los almacenen, protejan y utilicen de manera que maximicen su utilidad y accesibilidad, manteniendo siempre la seguridad y la privacidad.

Capacidad y Escalabilidad:

- La capacidad de almacenamiento debe ser evaluada basándose en el volumen y la frecuencia de los datos recogidos por la estación meteorológica. Esto incluye considerar la cantidad de datos generados por los sensores, como temperatura, humedad, velocidad del viento, etc., y la frecuencia con la que se recogen estos datos.
- La escalabilidad es crucial para acomodar el aumento de datos a lo largo del tiempo. Se debe considerar un sistema de almacenamiento que permita ampliar su capacidad sin interrumpir el funcionamiento de la estación meteorológica, lo que puede implicar soluciones de almacenamiento modular o basadas en la nube.

Integridad y Fiabilidad:

- La integridad de los datos se refiere a la precisión y consistencia de los datos a lo largo del tiempo. Es importante implementar mecanismos que verifiquen la exactitud de los datos recopilados, identifiquen posibles errores o corrupción de datos y los corrijan.
- La fiabilidad se asegura mediante sistemas de respaldo, como copias de seguridad periódicas y redundancia de datos. Estos sistemas son fundamentales para prevenir la pérdida de datos debido a fallos del hardware, errores de software o desastres naturales.

Accesibilidad y Recuperación de Datos:

- Los datos deben ser fácilmente accesibles para análisis y toma de decisiones. Esto incluye la implementación de interfaces y protocolos que permitan a los usuarios y sistemas acceder a los datos de manera eficiente y segura.
- La recuperación de datos es igualmente importante. Se deben establecer procedimientos para restaurar rápidamente los datos en caso de pérdida, lo que puede incluir soluciones de almacenamiento redundante o sistemas de recuperación de desastres.

Almacenamiento en la Nube vs. Local:

- **Almacenamiento en la Nube:**
 - **Ventajas:** Ofrece mayor escalabilidad, acceso remoto a los datos, y menores costos iniciales de infraestructura. La gestión del almacenamiento suele ser más sencilla, y los proveedores de servicios en la nube ofrecen robustas medidas de seguridad.
 - **Desventajas:** Dependencia de la conexión a Internet, preocupaciones potenciales sobre la seguridad y privacidad de los datos, y posibles costos recurrentes más elevados.
- **Almacenamiento Local:**
 - **Ventajas:** Control total sobre los datos y su seguridad, acceso rápido y directo a los datos, y no depende de una conexión a Internet.
 - **Desventajas:** Requiere una inversión inicial más alta en hardware y mantenimiento, limitaciones de escalabilidad y posibles desafíos en la implementación de medidas de seguridad avanzadas.

8.7. Normativas en pruebas

IEEE 1012

Norma para la verificación y validación de sistemas y programas informáticos. Especifica los procesos de verificación y validación (V&V) utilizados para determinar si los productos de desarrollo de una actividad dada se ajustan a los requisitos de dicha actividad y si el producto satisface su uso previsto y las necesidades del usuario.

IEEE 1220

Norma para la aplicación y gestión del proceso de ingeniería de sistemas. Define las tareas interdisciplinarias necesarias a lo largo del ciclo de vida de un sistema para transformar las necesidades, requisitos y limitaciones de las partes interesadas en una solución del sistema (Doran, 2006).

IEEE 829

Norma para la documentación de pruebas de software y sistemas. Especifica la forma de un conjunto de documentos para su uso en ocho etapas definidas de pruebas de software y pruebas de sistemas, cada etapa puede producir su propio tipo de documento (*IEEE Standard for System and Software Verification and Validation*, 2012).

8. METODOLOGÍA

La Figura 8 ilustra la secuencia metodológica adoptada en el proyecto de Desarrollo de un sistema de monitoreo meteorológico de bajo costo basado en tecnología IoT. El proceso inicia con la identificación de variables meteorológicas clave, seguida por la selección de hardware y componentes apropiados para la medición precisa de dichas variables. Posteriormente, se aborda la gestión de energía eficiente, lo que es crucial para la sostenibilidad del sistema, y se asegura una conexión estable a la fuente de alimentación. Con estos fundamentos establecidos, el sistema avanza hacia la conexión a Internet y la transmisión de datos, permitiendo el almacenamiento y visualización en la nube de la información recopilada. La interfaz de monitoreo meteorológico se diseña con el fin de proporcionar una visualización intuitiva y accesible de los datos.

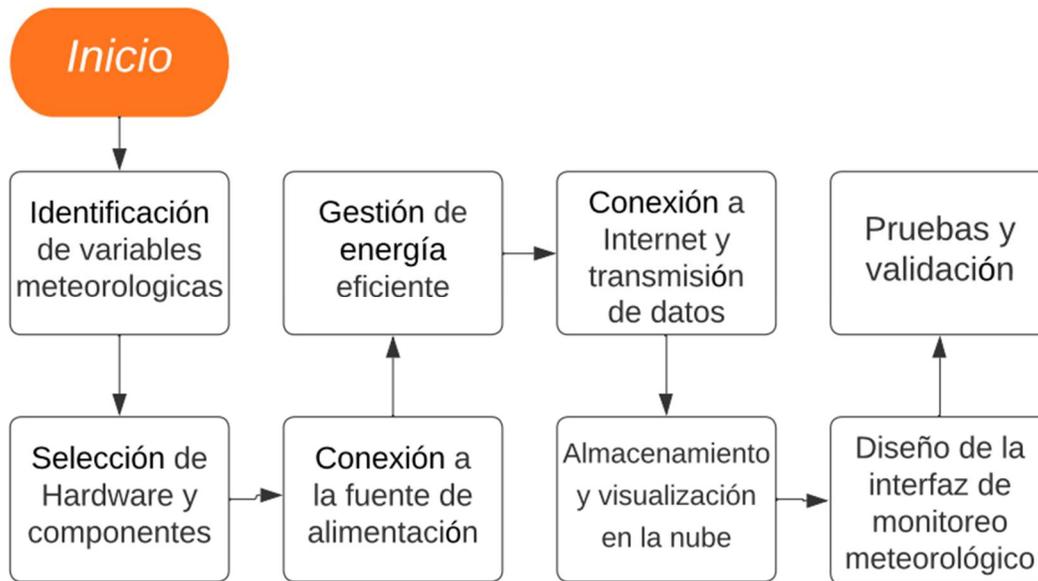


Figura 8 Diagrama de bloques de la Metodología propuesta.

Referente a la Figura 8 , se procede con la identificación meticulosa de las variables meteorológicas pertinentes, que incluyen, pero no se limitan a, temperatura, humedad, presión atmosférica y la cinética del viento. Se realiza la selección de hardware y componentes, en la cual el microcontrolador ESP8266 se distingue por sus capacidades de procesamiento y su aptitud para la conectividad, respaldado por sensores precisos y el anemómetro Davis Pro, para asegurar la exactitud en la captura de datos así como el sensor BME680 para la adquisición de las temperatura, humedad, presión y altitud.

Una vez establecido el hardware, se configura un centro de carga vinculado a paneles solares y baterías LIPO, estableciendo así un sistema de suministro de energía autónomo. Se enfatiza la gestión eficiente del consumo energético mediante la adecuación del microcontrolador a un modo de bajo consumo en periodos de inactividad. Lo siguiente es la etapa de conectividad, estableciendo una conexión a Internet y la transmisión de datos hacia la plataforma Ubidots en la nube, lo que facilita el almacenamiento y la visualización gráfica de las mediciones. La visualización de datos se facilita a través de una interfaz de usuario web, que permite una interpretación intuitiva y una supervisión efectiva de las condiciones meteorológicas.

El proceso culmina con una etapa de pruebas y validación, utilizando como referencia bases de datos de estaciones meteorológicas aledañas para corroborar la fiabilidad y precisión de la estación desarrollada.

8.1. Identificación de variables meteorológicas.

Se ha llevado a cabo una investigación exhaustiva sobre las variables climatológicas más relevantes, determinando que aspectos como la presión atmosférica, la humedad relativa, la temperatura, la altitud, así como la velocidad y dirección del viento, son fundamentales para comprender y analizar el clima de una región específica.

La presión atmosférica, que influye en los patrones meteorológicos, es crucial para identificar las condiciones climáticas predominantes. La humedad relativa, por otro lado, proporciona información valiosa sobre la cantidad de vapor de agua presente en el aire, lo que es vital para entender los fenómenos de precipitación y la comodidad ambiental. La temperatura ambiental es el indicador climático más directo y perceptible, afectando no solo a los ecosistemas y la biodiversidad, sino también a la actividad humana. La altitud juega un rol significativo en la variación de estas variables, dado que afecta tanto a la temperatura como a la presión atmosférica. Finalmente, la velocidad y dirección del viento son determinantes en la dispersión de polen y semillas, la formación de tormentas, y en la modulación de temperaturas a nivel local y regional.

Para corroborar y enriquecer este análisis, se ha consultado información proveniente del Servicio Meteorológico Nacional (SMN), cuyos datos y estudios aportan una perspectiva autorizada y actualizada sobre las condiciones climáticas y meteorológicas. La verificación cruzada de la información con fuentes del SMN asegura no solo la precisión y relevancia de los datos seleccionados, sino también su aplicabilidad en contextos tanto académicos como prácticos en el campo de la meteorología y climatología.

8.2. Selección de Hardware y Componentes

En el desarrollo del proyecto, se realizó una selección cuidadosa de sensores y microcontroladores para medir variables meteorológicas clave como la humedad,

temperatura y presión atmosférica. Tras una evaluación detallada, se decidió utilizar un sensor MEMS, específicamente el modelo BME680, por sus ventajas en términos de tamaño reducido y bajo consumo energético. Este sensor es conocido por su capacidad para realizar mediciones precisas de las variables mencionadas, lo cual es esencial para obtener datos fiables y exactos en el análisis meteorológico.

Para la medición de la dirección y velocidad del viento, se seleccionó el sistema Davis Vantage Pro. Este equipo incluye un anemómetro para medir la velocidad del viento y una veleta para determinar su dirección. La elección de este sistema se basó en su alta precisión y exactitud en las mediciones, características fundamentales para el estudio del comportamiento del viento y su impacto en el clima local.

En cuanto al microcontrolador, se optó por el modelo ESP8266, destacando por su bajo consumo energético de fábrica, que puede reducirse aún más mediante el uso de modos de bajo consumo. En este proyecto, se implementó la función de "deepsleep" entre ciclos de medición, lo que permite minimizar aún más el consumo de energía sin comprometer la eficiencia en la recopilación de datos. Esta característica es particularmente valiosa para asegurar una operación prolongada y sostenible del sistema de monitoreo meteorológico, especialmente en aplicaciones que requieren autonomía y eficiencia energética.

La Tabla 1 muestran las características más relevantes de los sensores atmosféricos usados.

Característica	Especificación
Dimensiones	3.0 x 3.0 x 0.93 mm, metal lid LGA package
Interfaz	I ² C (up to 3.4 MHz), SPI (up to 10 MHz)
Tensión de alimentación	VDD: 1.71 V - 3.6 V, VDDIO: 1.2 V - 3.6 V
Consumo de corriente	2.1 - 3.7 μ A (mode dependent), 0.15 μ A (sleep)
Rango de funcionamiento	-40 to +85 °C, 0 - 100% r.H., 300 - 1100 hPa
Independencia del sensor	Yes
Tiempo de respuesta del sensor de gas	< 1 s
Potencia del sensor de gas	< 0.1 mA (ultra-low power mode)
Salida del sensor de gas	IAQ index output
Respuesta del sensor de humedad	~8 s
Precisión de la humedad	\pm 3% r.H.
Histéresis de humedad	\pm 1.5% r.H.
Ruido del sensor de temperatura	0.12 Pa
Coeficiente Temp. Coeficiente	\pm 1.3 Pa/K

Tabla 1 Valores Sensor BME680

8.3. Conexión a la Fuente de Alimentación

Para garantizar la operatividad de la estación meteorológica en zonas donde no hay acceso a la red eléctrica, se implementó un sistema de alimentación autónomo. Se eligió una batería de polímero de litio (LiPo) debido a su eficiencia energética y capacidad de almacenamiento. Esta batería está conectada a un centro de carga, que a su vez se vincula a paneles solares para asegurar una fuente de energía renovable y sostenible.

La integración de paneles solares es clave para mantener la batería cargada y proporcionar una fuente de energía constante y confiable. Esta configuración es especialmente útil en entornos remotos o aislados donde el suministro eléctrico convencional no está disponible o es inestable.

Además, considerando el bajo consumo energético del microcontrolador ESP8266, seleccionado por su eficiencia en el uso de energía, se minimiza el riesgo de agotamiento rápido de la batería. Esto asegura que la estación meteorológica pueda funcionar de manera continua y eficaz en áreas sin conexión eléctrica. La

combinación de una batería LiPo, un sistema de carga solar, y un microcontrolador de bajo consumo, crea una solución robusta y autónoma, adecuada para la implementación de la estación meteorológica en una amplia gama de ubicaciones, incluyendo aquellas que presentan desafíos en términos de acceso a fuentes de energía tradicionales.

8.4. Gestión de Energía Eficiente

El microcontrolador ESP8266, en su funcionamiento normal, tiene un consumo de energía de aproximadamente 70 mA. Al entrar en el modo *deepsleep*, este consumo se reduce drásticamente a solo 100 μ A, que es equivalente a 0.1 mA. Esta reducción representa una disminución de aproximadamente el 99.86% en el consumo de energía.

En el contexto del proyecto, el microcontrolador está programado para realizar mediciones cada 10 minutos. Cada ciclo de medición dura alrededor de 10 segundos. Durante este breve periodo, el dispositivo opera a su consumo normal (70 mA), y por el resto del tiempo, permanece en el modo *deepsleep*, consumiendo solo 0.1 mA. Esta estrategia de gestión de energía es altamente eficiente, permitiendo que el sistema mantenga un consumo energético muy bajo en general. Esta eficiencia es crucial para asegurar la operatividad a largo plazo de la estación meteorológica, especialmente en ubicaciones remotas donde se depende de fuentes de energía renovables como los paneles solares. La combinación del bajo consumo del ESP8266 en *deepsleep* y su operación intermitente optimiza el uso de la batería y maximiza la viabilidad del sistema en entornos aislados.

Complementando esta eficiencia energética, se empleó la energía solar como fuente de alimentación principal. La combinación del modo *deepsleep* del ESP8266 con la energía renovable de los paneles solares asegura una operación sostenible y continua de la estación meteorológica, incluso en zonas sin acceso a la red eléctrica. Esta integración de tecnologías de bajo consumo y fuentes de energía renovables permite que la estación funcione de manera autónoma y eficiente en una amplia gama de entornos.

8.5. Conexión a Internet y Transmisión de Datos

A continuación, se muestra en la Ilustración 1 el pseudocódigo utilizado para la conexión a internet y la transmisión de datos:

Algoritmo Conexión_Sensor_Ubidots

```
Incluir <Ubidots.h>
Incluir <Wire.h>
Incluir <SPI.h>
Incluir <Adafruit_Sensor.h>
Incluir "Adafruit_BME680.h"

Definir SEALEVELPRESSURE_HPA Como 1013.25

Definir UBIDOTS_TOKEN Como Cadena // Token para Ubidots
Definir WIFI_SSID Como Cadena // SSID de la red Wi-Fi
Definir WIFI_PASS Como Cadena // Contraseña de la red Wi-Fi

Definir analogInPin Como Entero // Pin analógico de entrada
Definir sensorMaxValue Como Entero // Valor máximo del sensor
Definir degreesPerUnit Como Real // Grados por cada unidad del sensor
Definir sensorValue Como Entero

Definir interruptPin Como Byte // Pin para interrupción
Definir Debounce_Timer, Current_Event_Time, Last_Event_Time, Event_Counter Como Entero sin signo
Definir WindSpeed Como Real

Definir ubidots con UBIDOTS_TOKEN
Definir bme para I2C

Proceso setup()
  Iniciar Serial a 115200 bps
  Conectar Wi-Fi con WIFI_SSID y WIFI_PASS
  Verificar conexión Serial
  Imprimir "BME680 async test"

  Si no se inicializa bme entonces
    Imprimir error y detener ejecución
  Fin Si

  Configurar bme con parámetros de calibración

  Configurar pin de interrupción para WindSpeed
  Deshabilitar interrupciones durante la configuración
  Asignar interrupción al pin con función WSpeed_ISR
  Iniciar y configurar temporizador
  Habilitar interrupciones
Fin Proceso
```

Ilustración 1 Pseudocódigo inicialización de sensores.

Para establecer la conexión a Internet y transmitir los datos recopilados por la estación meteorológica, se implementó una estrategia basada en el uso de librerías especializadas de Ubidots. Estas librerías facilitan la conexión segura y eficiente con la base de datos en la nube, permitiendo el almacenamiento y análisis posterior de los datos.

```

Proceso loop()
  Leer sensorValue de analogInPin
  Calcular grados a partir de sensorValue

  Si grados es menor que 0 entonces
    Ajustar grados sumando 360
  Fin Si

  Comenzar lectura de bme y verificar éxito
  Esperar un momento para tareas paralelas
  Finalizar lectura de bme y verificar éxito
  Esperar antes de enviar datos

  Añadir datos de bme a Ubidots
  Añadir grados y WindSpeed a Ubidots
  Enviar datos a Ubidots y verificar si fue exitoso

  Si el envío fue exitoso entonces
    Imprimir confirmación
  Fin Si

  Desconectar Wi-Fi y poner ESP en modo de sueño profundo
Fin Proceso

Interrupción WSpeed_ISR()
Validar y actualizar el contador de eventos para WindSpeed
Fin Interrupción

Interrupción Timer_ISR()
Restablecer temporizador
Calcular WindSpeed
Reiniciar contador de eventos
Fin Interrupción

FinAlgoritmo

```

Ilustración 2 Pseudocódigo recolección de datos y transmisión a la nube.

El código comienza con la inclusión de librerías esenciales, como Ubidots para la conexión con la base de datos, y Adafruit_Sensor y Adafruit_BME680 para la interacción con los sensores. Se definen constantes y variables para configurar la conexión a Internet y los parámetros de los sensores. Dentro de la función setup, se establece la conexión Wi-Fi y se inicia el sensor BME680, que juega un papel crucial en la medición de variables ambientales como temperatura, humedad, presión y calidad del aire. De igual forma se implementó la función 'millis()' para conocer el tiempo transcurrido entre mediciones.

El sensor BME680 se inicializa con configuraciones específicas para el muestreo y la calibración de los datos. Durante el ciclo *loop*, se realizan lecturas periódicas del sensor, capturando datos ambientales precisos y confiables. Estos datos se almacenan temporalmente y luego se envían a Ubidots utilizando la función `'add()'`.

Además de la recopilación de datos ambientales, el código también incluye una funcionalidad para medir la dirección y velocidad del viento. Se utiliza un anemómetro conectado al pin de interrupción del microcontrolador. A través de una rutina de servicio de interrupción (ISR), se capturan y procesan las señales del anemómetro, lo que permite calcular la velocidad del viento. La dirección del viento se determina mediante un sensor conectado al pin analógico A0, que mide la orientación de la veleta.

Una vez que los datos de los sensores, incluyendo la velocidad y dirección del viento, se han recopilado y procesado, se añaden a Ubidots para su posterior transmisión. Tras la transmisión exitosa de los datos, el sistema se desconecta de la red Wi-Fi y entra en modo deepsleep durante 10 minutos, lo que permite conservar energía y garantiza una operación prolongada y eficiente.

8.6. Almacenamiento y Visualización en la Nube

La plataforma de Ubidots juega un papel importante en el almacenamiento y visualización de los datos recopilados por la estación meteorológica. Ubidots es una plataforma de Internet de las Cosas (IoT) diseñada específicamente para transformar los datos del sensor en información útil y accesible. Ofrece una solución integral para el almacenamiento de datos en la nube, así como herramientas avanzadas para la visualización y análisis de estos datos.

El funcionamiento de Ubidots se basa en la recepción y procesamiento de datos enviados por dispositivos IoT, como nuestra estación meteorológica. Una vez que los datos son transmitidos a Ubidots, a través de una conexión segura y utilizando las librerías específicas de Ubidots en el código de la estación, se almacenan en la plataforma. Este almacenamiento en la nube garantiza no solo la

seguridad y la integridad de los datos, sino también su disponibilidad constante para el acceso y análisis.

Para optimizar la visualización y comprensión de los datos recopilados, se eligió emplear los tableros de control (*dashboards*) de Ubidots. Esta herramienta permite la presentación gráfica de las variables climatológicas en tiempo real, facilitando su interpretación intuitiva y detallada.

Los *dashboards* se configuran para mostrar gráficos que representan las mediciones de temperatura, humedad, presión atmosférica, entre otras, permitiendo a los usuarios observar no solo los valores actuales, sino también sus tendencias y fluctuaciones a lo largo del tiempo. Esta capacidad de visualización temporal ofrece una visión clara y completa de la evolución del clima, destacando patrones, tendencias y anomalías que podrían pasar desapercibidas en un análisis más estático.

Además, el uso de gráficos interactivos en los *dashboards* enriquece la experiencia del usuario, permitiendo la exploración de los datos a través de diferentes escalas temporales y facilitando el análisis comparativo entre distintas variables climatológicas. Esta funcionalidad es esencial para una evaluación precisa del clima y sus variaciones, siendo una herramienta valiosa para investigadores, meteorólogos y cualquier interesado en el estudio y monitoreo del clima.

8.7. Diseño de la Interfaz de Monitoreo Meteorológico

Para facilitar el monitoreo meteorológico y garantizar su accesibilidad desde cualquier dispositivo con conexión a Internet, se desarrolló una página web específica. Esta página, diseñada y construida con HTML y CSS, ofrece una interfaz de usuario intuitiva y estéticamente agradable, adaptándose a diferentes tamaños de pantalla y dispositivos para una óptima experiencia de usuario, la implementación se encuentra en los Anexos B y C respectivamente.

La elección de HTML para la estructura de la página web asegura una base sólida y flexible, mientras que CSS se empleó para definir y personalizar los estilos, garantizando una presentación visual coherente y atractiva. Esta combinación

proporciona una plataforma robusta y fácil de usar para la visualización de datos meteorológicos.

Además, se integraron iframes embebidos de Ubidots en la página web. Esta característica de Ubidots permite insertar dashboards de manera directa en la página, facilitando la visualización de los datos recopilados. Los *iframes* proporcionan una ventana interactiva a los tableros de control de Ubidots, lo que permite a los usuarios ver los datos meteorológicos en tiempo real y con un alto grado de detalle.

La inclusión de estos *iframes* embebidos en la página web significa que los usuarios pueden acceder a información actualizada y detallada sobre las condiciones meteorológicas sin salir del entorno web. Esta integración mejora significativamente la experiencia del usuario, ofreciendo una visualización de datos fluida y coherente.

8.8. Pruebas y validación.

Para la fase de pruebas y validación del sistema, se realizó un análisis comparativo utilizando datos de estaciones meteorológicas adyacentes. A pesar de la ausencia de estaciones con datos públicos en el municipio de San Juan del Río, Querétaro, se seleccionaron las estaciones más próximas ubicadas en Huichapan y Huimilpan para este propósito. Los datos para la comparación fueron obtenidos de la base de datos pública del Servicio Meteorológico Nacional (SMN), proporcionando así un marco de referencia para evaluar la precisión y el rendimiento del sistema meteorológico desarrollado.(CONAGUA, 2005)

Los análisis realizados incluyen una serie de gráficos y tablas para evaluar y comparar las mediciones meteorológicas:

1. **Gráfico de Líneas Temporales:** Muestra la temperatura a lo largo del tiempo, comparando datos de las estaciones meteorológicas de Huichapan, Huimilpan y San Pedro Ahuacatlán.
2. **Tabla de Temperaturas Máximas:** Presenta un cuadro comparativo de las temperaturas máximas registradas por una estación privada en

San Juan del Río y una estación denominada "San Pedro Ahuacatlan", incluyendo las diferencias de temperatura observadas.

3. **Gráficos de Caja:** Ilustran la distribución estadística de las variables meteorológicas, como temperatura, humedad y presión, comparando las tres estaciones mencionadas.

Dichos análisis se realizaron usando Python y las librerías pandas, matplotlib y seaborn.

El pseudocódigo presentado en la imagen proporciona una estructura lógica para un algoritmo de análisis de datos climáticos, la implementación se encuentra en el Anexo D. A continuación, se ofrece una explicación detallada del proceso que sigue este pseudocódigo:

Algoritmo Analisis_Datos_Climaticos

```
# Importar librerías
Importar pandas como pd
Importar matplotlib.pyplot como plt
Importar seaborn como sns

# Cargar y limpiar los datos de San Pedro Ahuacatlan
Definir data_san_pedro como pd.leer_csv('SanPedroAhuacatlan.csv', codificación='ISO-8859-1')
Definir data_san_pedro_cleaned como data_san_pedro.eliminar(index=rango(8))
Establecer columnas de data_san_pedro_cleaned como ['timestamp', 'temperature', 'pressure', 'humidity']
data_san_pedro_cleaned = data_san_pedro_cleaned desde el segundo elemento hasta el final
Convertir data_san_pedro_cleaned['timestamp'] a formato de fecha y hora
Convertir data_san_pedro_cleaned['temperature'], ['pressure'], ['humidity'] a numérico

# Función para limpiar datos de las estaciones
Funcion Limpiar_Datos_Estacion(datos)
    datos_limpios = datos.eliminar(index=rango(8))
    Establecer columnas de datos_limpios como ['timestamp', 'temperature', 'humidity', 'pressure']
    Convertir datos_limpios['timestamp'] a formato de fecha y hora con formato '%d/%m/%Y %H:%M'
    Convertir datos_limpios['temperature'], ['pressure'], ['humidity'] a numérico
    Retornar datos_limpios
FinFuncion
```

Ilustración 3 Pseudocódigo Limpieza de datos.

En la Ilustración 3 se presenta la lógica aplicada para la depuración de los datos recabados de las distintas estaciones, previo a su análisis correspondiente. Esta etapa de procesamiento de datos es fundamental para optimizar y facilitar la interpretación de los resultados, asegurando así una mayor precisión y fiabilidad en el análisis siguiente.

```

# Combinar los datos para análisis comparativo
Etiquetar cada conjunto de datos con su respectiva estación
Combinar todos los conjuntos de datos limpios en uno solo y eliminar los valores nulos

# Visualizar distribuciones
Preparar figura para gráficos
    Crear diagramas de caja para temperatura, presión y humedad usando seaborn
    Mostrar gráficos

    # Análisis de fiabilidad
    Obtener resumen estadístico de la temperatura, presión y humedad de San Pedro
    Calcular el rango intercuartílico (IQR) y determinar valores atípicos
    Obtener resumen comparativo de las medias de temperatura, presión y humedad por estación
    Analizar intervalos de tiempo entre mediciones y describir irregularidades

    # Combinar los datos de las tres estaciones
    Combinar todos los conjuntos de datos limpios en uno solo, ignorar índices antiguos

    # Convertir todas las marcas de tiempo a tz-naive
    Copiar el conjunto de datos combinado
    Convertir la marca de tiempo de cada entrada a tz-naive

    # Encontrar el punto de inicio común más temprano entre las tres estaciones
    Definir start_date_naive como el mínimo de las marcas de tiempo tz-naive

    # Filtrar los datos para incluir solo mediciones desde el punto de inicio común
    Filtrar los datos para incluir entradas desde start_date_naive

        # Crear la gráfica con el mismo punto de inicio
        Preparar figura para la gráfica
            Graficar la temperatura para cada estación con el mismo punto de inicio
            Configurar título, etiquetas, leyenda, rotación de marcas de tiempo y cuadrícul
            Mostrar gráfica

FinAlgoritmo

```

Ilustración 4 Pseudocódigo análisis climático.

El pseudocódigo presentado en la Ilustración 4 proporciona una estructura lógica para un algoritmo de análisis de datos climáticos. A continuación, se ofrece una explicación detallada del proceso que sigue este pseudocódigo:

Cargar y Limpiar Datos de Otras Estaciones:

Se cargan y limpian datos de otras estaciones meteorológicas utilizando una función definida para este propósito. El proceso implica leer archivos CSV para cada estación y aplicar la función de limpieza '*Limpiar_Datos_Estacion*' para normalizar la estructura de los datos y hacerlos coherentes para su posterior análisis.

Agregar Información de Ubicación:

A cada conjunto de datos limpios se agrega información de latitud, longitud y altitud. Esta información es crucial para análisis geospaciales y para identificar las correlaciones entre la ubicación geográfica y las variables climáticas.

Combinar los Datos para Análisis Comparativo:

Se etiqueta cada conjunto de datos con el nombre de su respectiva estación y luego se combinan todos los conjuntos en uno solo. Los valores nulos se eliminan para garantizar la integridad del análisis comparativo.

Visualizar Distribuciones:

Se prepara la figura para gráficos y se crean diagramas de caja para temperatura, presión y humedad usando seaborn. Estos gráficos son útiles para visualizar la distribución de los datos y detectar outliers o tendencias.

Análisis de Fiabilidad:

Se realiza un análisis estadístico de los datos de San Pedro para obtener un resumen de las variables climáticas, se calcula el rango intercuartílico (IQR) y se identifican los valores atípicos. También se hace un resumen comparativo por estación y se analizan los intervalos de tiempo entre mediciones para encontrar posibles irregularidades.

Combinar Datos de las Tres Estaciones:

Todos los conjuntos de datos limpios se combinan en uno solo, y los índices antiguos son ignorados para crear un conjunto de datos unificado.

Convertir Marcas de Tiempo a tz-naive:

Se copia el conjunto de datos combinado y se convierten las marcas de tiempo de cada entrada a un formato sin información de zona horaria (tz-naive).

Encontrar el Punto de Inicio Común Más Temprano:

Se busca el punto de inicio más temprano común entre las tres estaciones para sincronizar los datos.

Filtrar los Datos para Incluir Solo Mediciones desde el Punto de Inicio Común:

Los datos se filtran para incluir solo las mediciones que se han realizado desde el punto de inicio común identificado anteriormente.

Crear la Gráfica con el Mismo Punto de Inicio:

Se prepara una nueva figura y se grafica la temperatura para cada estación comenzando desde el mismo punto de inicio. Se configuran los títulos, etiquetas, leyenda, la rotación de las marcas de tiempo y la cuadrícula. Finalmente, se muestra la gráfica, lo cual proporciona una comparación visual directa de las tendencias de temperatura a lo largo del tiempo entre las diferentes estaciones.

9. RESULTADOS

A continuación, se presentan los resultados obtenidos al seguir la metodología propuesta. En esta sección, se detallará el proceso requerido para acceder, visualizar y analizar los datos recabados, asegurando su fiabilidad y precisión. Para comenzar, es necesario visitar la página web alojada en el siguiente enlace: "<https://furulango.github.io/>". El acceso a esta página se facilita a través de cualquier navegador web moderno y compatible. Una vez en la página, los usuarios podrán observar los datos meteorológicos recopilados de manera interactiva y detallada.

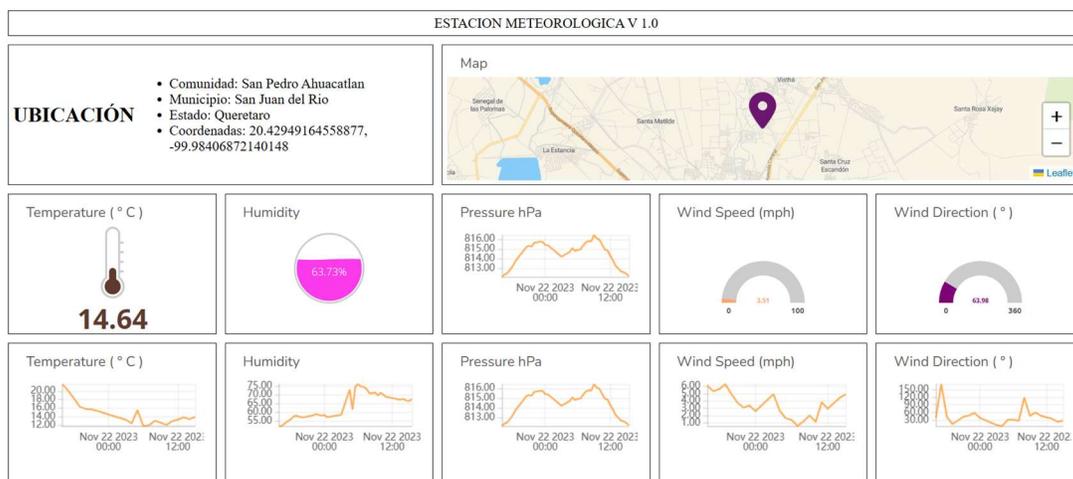


Figura 9 Interfaz Web estación meteorológica.

En la Figura 9 se ilustra la distribución de los distintos valores climatológicos, así como su localización geográfica exacta, marcada en coordenadas y representada en un mapa interactivo. En esta visualización, las variables climatológicas recopiladas se muestran de dos maneras distintas: por un lado, se exhibe el último valor registrado para cada variable, y por otro, se presenta una gráfica que traza la evolución de estos valores a lo largo del tiempo. Esta representación dual facilita

una comprensión más completa tanto del estado actual del clima como de sus tendencias y cambios a lo largo de un período determinado.



Ilustración 5 Estación meteorológica.

La Ilustración 5 muestra la instalación de la estación meteorológica. Se observa el anemómetro en la parte superior para registrar la velocidad y dirección del viento, junto a los sensores ubicados de manera interna en la caja de mediciones.

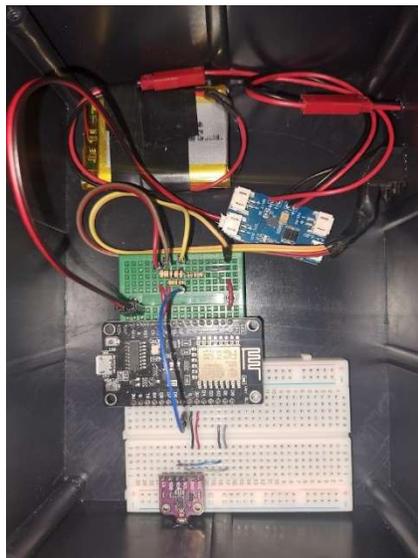


Ilustración 6 Conexión sistema de medición.

En la Ilustración 6 adjunta se muestra la conexión del sistema de medición, que consta del ESP8266, el sensor BME680, la conexión del anemómetro, el centro de carga y la batería LIPO de 6500mAh.

```
Sending data...
Datos enviados.
Tiempo para mediciones y transmisión: 4448 ms
```

Figura 10 Tiempo de medición y transmisión por ciclo.

Para validar la eficiencia en el uso de la energía del microcontrolador ESP8266, se realizaron mediciones del consumo energético durante la captura y transmisión de datos. Se registró un consumo medio de 75 mA durante la fase activa de ejecución y un consumo promedio de 10 μ A en el modo de reposo (*DeepSleep*). La frecuencia de operación se estableció en ciclos de 10 minutos, con una duración activa promedio de 5 segundos por ciclo, dicha medición se llevó a cabo al utilizar la función '*millis()*' que nos permite obtener el número de milisegundos transcurridos desde que el programa empezó su ejecución y dado que el programa tiene ciclos cada 10 minutos nos permite obtener el tiempo exacto que le toma al microcontrolador realizar la medición y transmisión de los datos. Para calcular el consumo energético promedio por ciclo, se aplicó la siguiente fórmula:

Consumo Promedio x ciclo (mA)

$$= \frac{\text{Consumo activo} * \text{Tiempo activo} + \text{Consumo DeepSleep} * \text{Tiempo DeepSleep}}{\text{Tiempo activo} + \text{Tiempo DeepSleep}}$$

Ecuación 1 Consumo promedio por ciclo.

Donde:

- Consumo en modo activo = 170 mA (durante 5 segundos)
- Consumo en modo DeepSleep = 0.01 mA (durante 10 minutos)

Aplicando la Ecuación 1, se obtiene un consumo promedio por ciclo de 1.41 mA. Para estimar la duración de la batería, se utilizó una batería LIPO con una capacidad de 6500 mAh.

$$\text{Duracion de la Bateria (horas)} = \frac{6500 \text{ mAh}}{\text{Consumo Promedio por ciclo (mA)}}$$

Ecuación 2 Duración de la batería.

La duración estimada de la batería, basada en el consumo promedio calculado de 1.41 mA, es de aproximadamente 4609.92 horas, lo que equivale a una autonomía de 192.08 días. Adicionalmente, la integración de paneles solares representa un componente innovador en el diseño del sistema. Estos paneles no solo prolongan la vida de la batería, sino que también promueven la sostenibilidad energética del dispositivo. Como menciona Mohammed et al.,(2019), la inclusión de fuentes de energía renovable puede aumentar significativamente la autonomía de dispositivos IoT, especialmente en entornos con adecuada exposición solar. En nuestro caso, la utilización de paneles solares posibilita un funcionamiento prácticamente ininterrumpido, lo que es esencial para aplicaciones críticas donde el reemplazo o recarga de baterías es logísticamente desafiante.

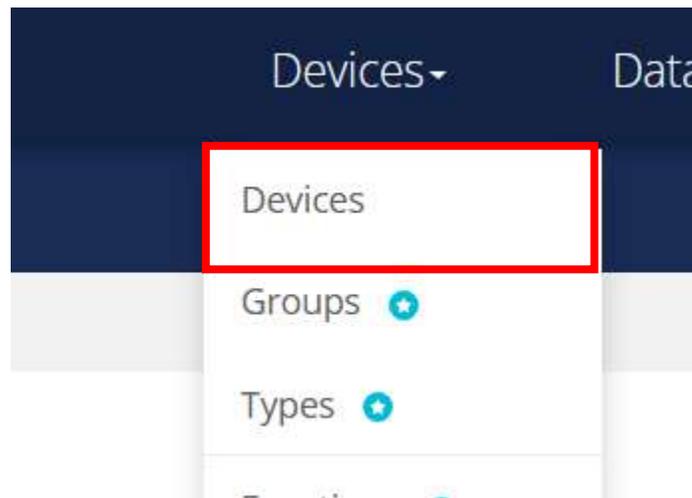


Figura 11 Selección de dispositivos.

Para realizar el análisis, la plataforma Ubidots ofrece la capacidad de descargar los datos climatológicos registrados en diversos intervalos de tiempo. Para acceder a esta opción, es necesario dirigirse a la sección 'Devices' en nuestra cuenta de Ubidots, tal como se detalla en la Figura 11 Selección de dispositivos. Allí, podremos seleccionar específicamente el dispositivo que corresponde a nuestra estación meteorológica. Esta herramienta resulta esencial para obtener una perspectiva

amplia y precisa del clima, permitiendo un análisis profundo y personalizado de los datos recopilados.

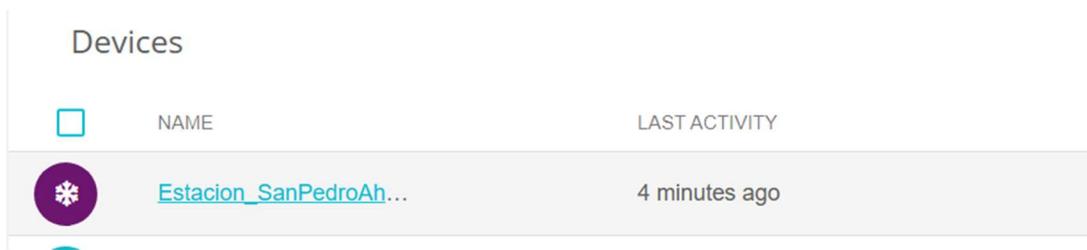


Figura 12 Selección estación San Pedro Ahuacatlán.

Dentro de la sección 'Devices' de la plataforma Ubidots, se nos brinda la posibilidad de explorar todos los dispositivos vinculados a nuestra cuenta. Para este análisis en particular, nos centraremos en el dispositivo denominado "Estacion_SanPedroAhuacatlan". Esta funcionalidad permite un acceso directo y eficiente a los datos específicos recopilados por la estación meteorológica de San Pedro Ahuacatlán, facilitando una revisión detallada y un manejo de la información climática relevante.

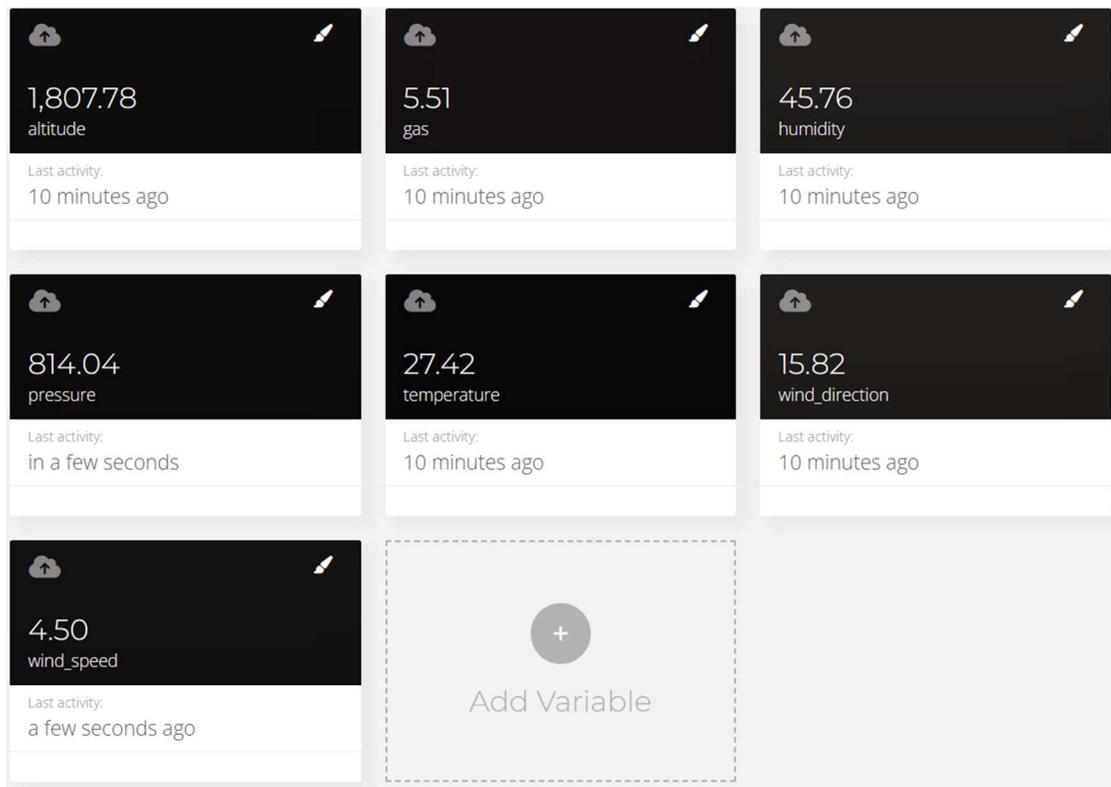


Figura 13 Variables climatológicas en Ubidots.

Al acceder a los detalles del dispositivo “Estacion_SanPedroAhuacatlan” en Ubidots, nos encontramos con un compendio de todos los datos transmitidos, tal como se muestra en la Figura 13. En este contexto, el elemento importante es la variable de la temperatura. Esta sección nos permite examinar en profundidad la información relativa a las mediciones de temperatura, proporcionando una visión clara y detallada de esta variable climática esencial. La disposición ordenada y clara de los datos facilita su análisis y comprensión, permitiendo una valoración precisa del comportamiento climatológico en la región estudiada.

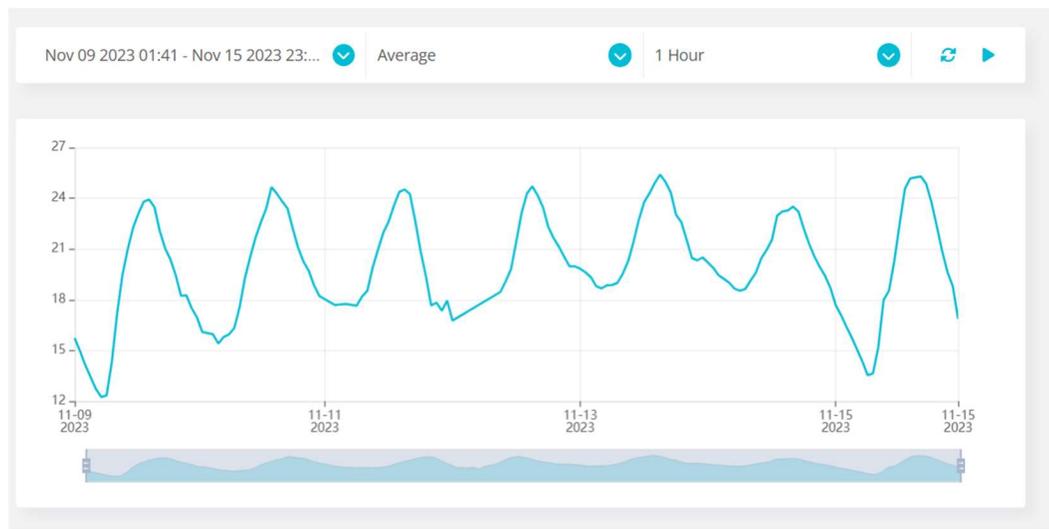


Figura 14 Grafica Ubidots temperatura.

Al explorar la variable de temperatura en la plataforma Ubidots, nos encontramos con una gráfica que traza el comportamiento de esta variable a lo largo de un periodo específico, tal y como se detalla en la Figura 14. Esta representación gráfica es una herramienta valiosa para visualizar y comprender las tendencias y fluctuaciones de la temperatura en el tiempo. Para proceder con la descarga de estos valores, es suficiente con dirigirse al icono de descarga como se indica en la Figura 15.

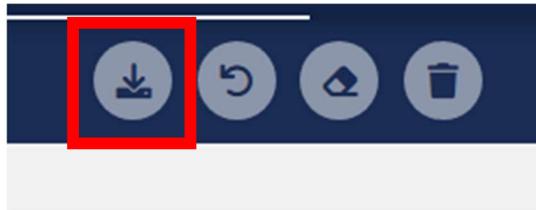


Figura 15 Sección de descarga.

A través de este icono, se nos ofrece la flexibilidad de seleccionar el intervalo de tiempo específico que nos interesa analizar, así como la opción de enviar la base de datos resultante directamente a nuestra dirección de correo electrónico así como el formato de la zona horaria. Esta funcionalidad simplifica enormemente el proceso de recopilación y análisis de datos, permitiendo un estudio personalizado y detallado de la información climatológica, dicha sección se muestra en la Figura 16.

temperature ×

Export data

Start date

End date

Recipients
A list of comma-separated emails.

Time zone

Variable identifier
Column name of the variable.

Columns

Figura 16 Descarga de base datos.

Se realizó la comparación de los valores de temperatura máxima respecto a la estación cercana a San Juan del Rio, Querétaro proporcionados por The Weather

Channel desde el 9 de noviembre al 15 de noviembre del 2023 y se obtuvo lo siguiente.

Fecha	Temperatura máxima San Pedro Ahuacatlán (°C)	Temperatura máxima Estación Privada San Juan del Rio(°C)	Diferencia de Temperatura (°C)
2023-11-09	22.96	25	2.04
2023-11-10	23.81	24	0.19
2023-11-11	23.69	24	0.31
2023-11-12	23.96	24	0.04
2023-11-13	24.49	24	0.49
2023-11-14	22.74	22	0.74
2023-11-15	24.94	24	0.94

Tabla 2 Comparativa temperaturas máximas

Consistencia en las Temperaturas: Las temperaturas máximas diarias de la estación meteorológica están bastante cercanas a las de la estación privada, con diferencias que varían desde tan solo 0.04°C hasta 2.04°C.

Variabilidad Normal: Es normal observar alguna variación en las lecturas entre diferentes estaciones meteorológicas, incluso cuando están relativamente cercanas entre sí. Esto puede deberse a microclimas locales, diferencias en la ubicación exacta, altitud y otros factores ambientales.

Día con Mayor Diferencia: El 9 de noviembre presenta la mayor diferencia (2.04°C), lo cual podría ser un punto de interés para investigar más. Factores como la sombra, la exposición al sol, o la proximidad a estructuras o cuerpos de agua podrían influir en esta variación.

Fiabilidad: En general, las temperaturas registradas muestran una buena consistencia con las de la estación privada. Esto indicaría que la estación está funcionando de manera fiable.

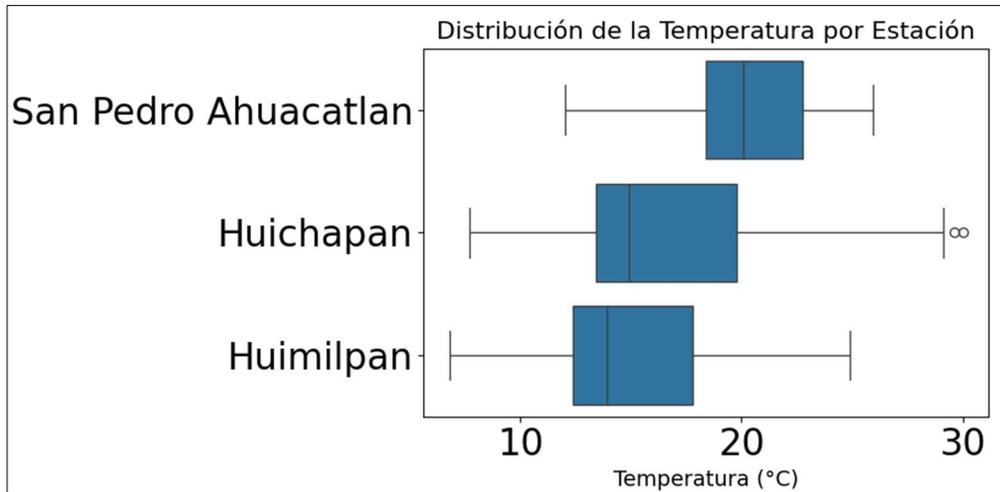


Figura 17 Grafica distribución temperatura.

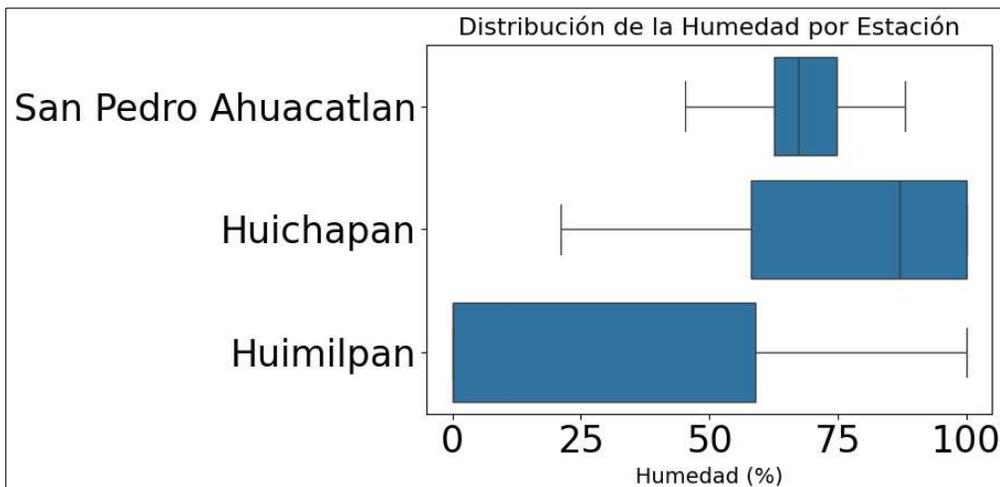


Figura 18 Grafica distribución humedad.

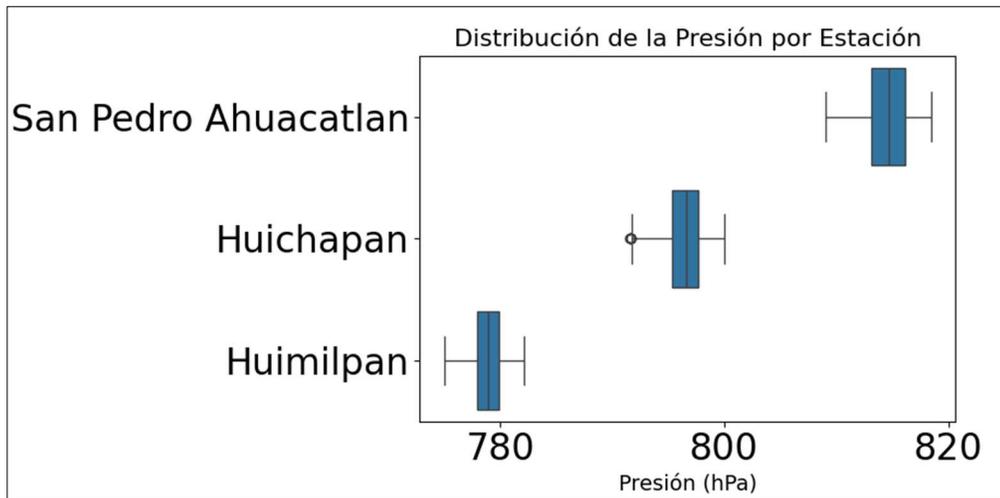


Figura 19 Grafica distribución presión.

Temperatura:

- San Pedro Ahuacatlán tiene la mayor variabilidad de temperatura con algunos valores extremadamente altos, lo que sugiere que esta estación podría experimentar una mayor influencia de factores que elevan la temperatura o que las mediciones se tomaron durante un período con fluctuaciones significativas de temperatura.
- Huichapan y Huimilpan tienen rangos de temperatura más estrechos, lo que podría indicar condiciones climáticas más estables o un período de medición donde las temperaturas no fluctuaron tanto como en San Pedro Ahuacatlán.

Presión:

- San Pedro Ahuacatlán y Huichapan tienen distribuciones de presión similares, lo que puede indicar que ambas estaciones están a altitudes similares o experimentan condiciones atmosféricas similares.
- Huimilpan muestra una variabilidad mucho más amplia en la presión y una mediana significativamente más baja. Esto es consistente con una estación que se encuentra a una mayor altitud, donde se esperarían lecturas de presión más bajas.

Humedad:

- San Pedro Ahuacatlán tiene una distribución amplia de humedad, lo que puede reflejar una variabilidad diurna significativa o una influencia de factores que afectan la humedad, como la proximidad al agua o la vegetación.
- Huichapan tiene una humedad relativamente alta y menos variabilidad, lo que sugiere condiciones más consistentemente húmedas durante el período de medición.
- Huimilpan muestra una amplia variabilidad con una mediana más baja, lo que podría indicar condiciones más secas en general, pero con algunos eventos que aportan humedad significativa.

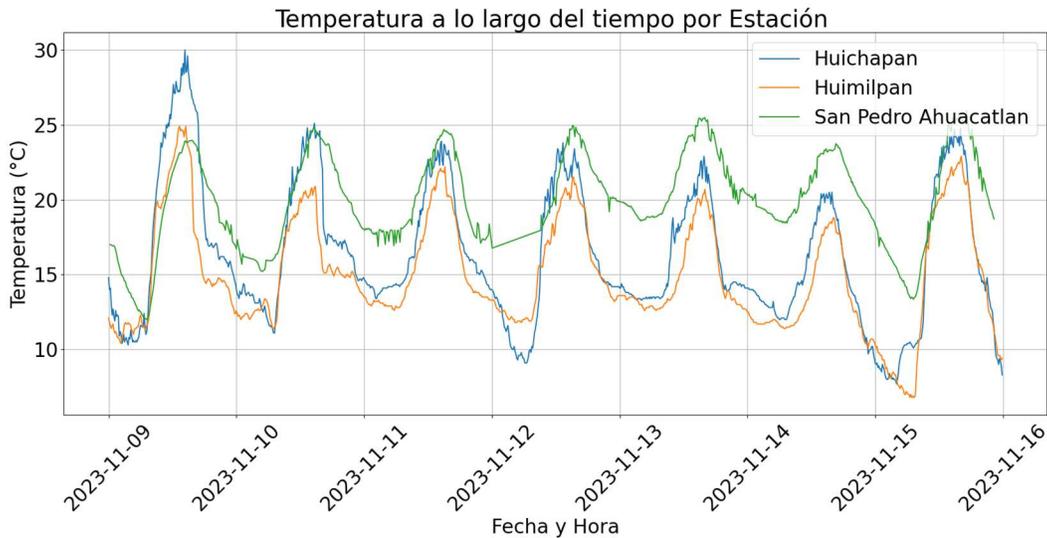


Figura 20 Grafica comparativa de temperatura entre las estaciones.

En la Figura 20, podemos observar lo siguiente:

- **Tendencias Temporales:** Las tres estaciones parecen seguir una tendencia temporal similar, con picos y valles que coinciden en el tiempo. Esto sugiere una influencia regional en la temperatura, como cambios en el sistema climático que afectan a todas las estaciones simultáneamente.
- **Variaciones de Temperatura:** Aunque las tendencias son similares, hay diferencias claras en los valores de temperatura entre las estaciones. Estas diferencias pueden atribuirse a la altitud, la ubicación geográfica y otros factores micro climáticos.
- **Eventos Sincrónicos:** Los picos de temperatura, que podrían indicar olas de calor o momentos del día con insolación máxima, parecen ocurrir al mismo tiempo en las tres estaciones, lo que refuerza la idea de un patrón climático compartido.
- **Amplitud Térmica:** Las diferencias en la amplitud de las fluctuaciones de temperatura entre las estaciones podrían indicar diferentes capacidades de retención de calor del entorno local o diferentes respuestas a la radiación solar debido a factores como la vegetación, la urbanización o la proximidad a cuerpos de agua.

9.1. Conclusiones

A lo largo de este proceso, se ha puesto especial énfasis en la selección de componentes, la optimización del consumo energético, y la implementación de una interfaz de usuario efectiva para la visualización de datos en tiempo real. Las conclusiones que se presentan a continuación sintetizan los resultados clave del proyecto, evalúan su impacto en el campo del monitoreo climático y proponen direcciones para futuras investigaciones.

- **Selección de Componentes y Construcción de la Estación Meteorológica:** Se logró con éxito la selección e integración de componentes para la construcción de una estación meteorológica de bajo costo. Los sensores y dispositivos seleccionados demostraron ser adecuados para medir variables climáticas esenciales, cumpliendo con el objetivo de monitorear eficazmente el clima.
- **Implementación de Tecnología IoT y Energía Solar:** La estación meteorológica incorpora eficientemente la tecnología del Internet de las Cosas (IoT), logrando una transmisión en tiempo real y almacenamiento de datos climáticos en la nube. La integración de una fuente de alimentación solar y una batería ha permitido un funcionamiento autónomo y sostenible, reduciendo significativamente la dependencia de fuentes de energía convencionales.
- **Eficiencia Energética y Autonomía Operativa:** Los resultados indican un consumo energético promedio por ciclo de 1.41 mA, con una duración estimada de la batería de aproximadamente 4609.92 horas (192.08 días). Esto valida la efectividad del modo de bajo consumo implementado y la viabilidad de la estación meteorológica para operaciones a largo plazo.
- **Precisión y Fiabilidad de los Datos:** La comparación de los datos recopilados con bases de datos de estaciones meteorológicas cercanas muestra una buena consistencia, con diferencias mínimas en las mediciones de temperatura. Esto sugiere que la estación meteorológica de bajo costo es una alternativa fiable y precisa para la recopilación de datos climáticos.

- **Desarrollo e Interfaz de Usuario:** La interfaz de usuario desarrollada proporciona una visualización clara y accesible de los datos climáticos en tiempo real. La facilidad de acceso y la presentación intuitiva de los datos a través de la plataforma web mejoran significativamente la experiencia del usuario.
- **Contribución al Monitoreo Climático y Sostenibilidad:** Este proyecto contribuye al campo del monitoreo climático ofreciendo una solución de bajo costo y energéticamente eficiente. La estación meteorológica demuestra ser una herramienta valiosa para la observación y el análisis del clima, con un potencial significativo para aplicaciones en educación, investigación y monitoreo ambiental.

9.2. Prospectivas

Para futuros trabajos, se recomienda explorar la expansión de la red de estaciones meteorológicas para cubrir un área geográfica más amplia, lo que permitiría un análisis más detallado de los patrones climáticos regionales. Además, sería beneficioso investigar la inclusión de sensores adicionales para medir otras variables climáticas relevantes como por ejemplo sensores de calidad del aire, radiación ultravioleta, entre otros, aumentando así la versatilidad y utilidad de la estación meteorológica.

REFERENCIAS

Abbass, K., Qasim, M. Z., Song, H., Murshed, M., Mahmood, H., & Younis, I. (2022). A review of the global climate change impacts, adaptation, and sustainable mitigation measures. *Environmental Science and Pollution Research*, 29(28), 42539-42559. <https://doi.org/10.1007/s11356-022-19718-6>

Bermeo, D., & Fernando, A. (2023). *Diseño de una red de comunicación inalámbrica para las estaciones meteorológicas en la isla San Cristóbal-Galápagos.*

Bourouina, T. (2021). Ubiquitous Sensing with MEMS-FTIR Spectroscopy Applications in Agri-Food and Environmental Monitoring. *2021 21st International Conference on Solid-State Sensors, Actuators and Microsystems (Transducers)*, 5-5.

<https://doi.org/10.1109/Transducers50396.2021.9495386>

Climate change 2021: The physical science basis: Summary for policymakers: Working group I contribution to the sixth Assessment report of the Intergovernmental Panel on Climate Change. (2021). IPCC.

CONAGUA. (2005). *Estaciones Meteorológicas Automáticas (EMA's).*

<https://smn.conagua.gob.mx/es/observando-el-tiempo/estaciones-meteorologicas-automaticas-ema-s>

Doran, T. (2006). IEEE 1220: For Practical Systems Engineering. *Computer*, 39(5), 92-94. <https://doi.org/10.1109/MC.2006.164>

Estévez, J., & Gavilán, P. (2008). *PROCEDIMIENTOS DE VALIDACIÓN DE DATOS DE ESTACIONES METEOROLÓGICAS AUTOMÁTICAS. APLICACIÓN A LA RED DE INFORMACIÓN AGROCLIMÁTICA DE ANDALUCÍA.*

Garcia, A. T. (2016). *Diseño e implementación de una estación meteorológica con Raspberry Pi.*

IEEE standard for system and software verification and validation. (2012). Institute of Electrical and Electronics Engineers.

- Ioannou, K., Karampatzakis, D., Amanatidis, P., Aggelopoulos, V., & Karmiris, I. (2021). Low-Cost Automatic Weather Stations in the Internet of Things. *Information*, 12(4), 146. <https://doi.org/10.3390/info12040146>
- Kodali, R. K., & Mahesh, K. S. (2016). A low cost implementation of MQTT using ESP8266. *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, 404-408. <https://doi.org/10.1109/IC3I.2016.7917998>
- Korolkov, V. A., Pustovalov, K. N., Tikhomirov, A. A., Telminov, A. E., & Kurakov, S. A. (2018). Autonomous weather stations for unmanned aerial vehicles. Preliminary results of measurements of meteorological profiles. *IOP Conference Series: Earth and Environmental Science*, 211, 012069. <https://doi.org/10.1088/1755-1315/211/1/012069>
- Kumar, U., Singh, S., Bisht, J. K., & Kant, L. (2021). Use of meteorological data for identification of agricultural drought in Kumaon region of Uttarakhand. *Journal of Earth System Science*, 130(3), 121. <https://doi.org/10.1007/s12040-021-01622-1>
- Li, W., Awais, M., Ru, W., Shi, W., Ajmal, M., Uddin, S., & Liu, C. (2020). Review of Sensor Network-Based Irrigation Systems Using IoT and Remote Sensing. *Advances in Meteorology*, 2020, 1-14. <https://doi.org/10.1155/2020/8396164>
- Lin, D., MacDonald, R., Calbaza, D., Popp, J., Johnson, T., Andarawis, E., & Aimi, M. (2021). Polaris—A Low Cost MEMS Fabrication Platform for Navigation-Grade Inertial Sensors. *2021 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, 1-4. <https://doi.org/10.1109/INERTIAL51137.2021.9430465>

- Malozyomov, B. V., Martyushev, N. V., Voitovich, E. V., Kononenko, R. V., Konyukhov, V. Yu., Tynchenko, V., Kukartsev, V. A., & Tynchenko, Y. A. (2023). Designing the Optimal Configuration of a Small Power System for Autonomous Power Supply of Weather Station Equipment. *Energies*, 16(13), 5046. <https://doi.org/10.3390/en16135046>
- Marcos Martín, C. (2018). Satélites meteorológicos. En C. Santos Burguete (Ed.), *Física del caos en la predicción meteorológica* (pp. 83-102). Agencia Estatal de Meteorología. <https://doi.org/10.31978/014-18-009-X.07>
- Martinez, E. G. (2022). Historia de la meteorología en la Norpatagonia chilena: La estación meteorológica “Juan Kalt Bode” (1935-2012). *História Unisinos*, 26.
- Math, R. K. M., & Dharwadkar, N. V. (2018). IoT Based Low-cost Weather Station and Monitoring System for Precision Agriculture in India. *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)* I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference On, 81-86. <https://doi.org/10.1109/I-SMAC.2018.8653749>
- Misra, N. N., Dixit, Y., Al-Mallahi, A., Bhullar, M. S., Upadhyay, R., & Martynenko, A. (2022). IoT, Big Data, and Artificial Intelligence in Agriculture and Food Industry. *IEEE Internet of Things Journal*, 9(9), 6305-6324. <https://doi.org/10.1109/JIOT.2020.2998584>
- Mohammed, N. Q., Ahmed, M. S., Mohammed, M. A., Hammood, O. A., Alshara, H. A. N., & Kamil, A. A. (2019). Comparative Analysis between Solar and Wind Turbine Energy Sources in IoT Based on Economical and Efficiency Considerations. *2019 22nd International Conference on Control Systems*

and Computer Science (CSCS), 448-452.

<https://doi.org/10.1109/CSCS.2019.00082>

Morón, C., Diaz, J., Ferrández, D., & Saiz, P. (2018). Design, Development and Implementation of a Weather Station Prototype for Renewable Energy Systems. *Energies*, 11(9), 2234. <https://doi.org/10.3390/en11092234>

Muñoz-Mandujano, M., Gutiérrez-López, M. A., & Vargas-Díaz, J. A. (2022). *Modelo y algoritmo de pronóstico CRHUDA aplicado a las estaciones de la red de monitoreo RedCIAQ*.

Peterson, T. C., Heim, R. R., Hirsch, R., Kaiser, D. P., Brooks, H., Diffenbaugh, N. S., Dole, R. M., Giovannetone, J. P., Guirguis, K., Karl, T. R., Katz, R. W., Kunkel, K., Lettenmaier, D., McCabe, G. J., Paciorek, C. J., Ryberg, K. R., Schubert, S., Silva, V. B. S., Stewart, B. C., ... Wuebbles, D. (2013). Monitoring and Understanding Changes in Heat Waves, Cold Waves, Floods, and Droughts in the United States: State of Knowledge. *Bulletin of the American Meteorological Society*, 94(6), 821-834. <https://doi.org/10.1175/BAMS-D-12-00066.1>

Rabiela, G. N. (2010). *Diseño e implementación de una estación meteorológica de manufactura nacional*.

Sabancı, K., Rusen, S. E., & Konuralp, A. (2019). *Design of a low cost automatic meteorological weather station*. 8.

Sepulcre-Canto, G., Horion, S., Singleton, A., Carrao, H., & Vogt, J. (2012). Development of a Combined Drought Indicator to detect agricultural drought in Europe. *Natural Hazards and Earth System Sciences*, 12(11), 3519-3531. <https://doi.org/10.5194/nhess-12-3519-2012>

Siems-Anderson, A. R., Walker, C. L., Wiener, G., Mahoney, W. P., & Haupt, S. E.

(2019). An adaptive big data weather system for surface transportation.

Transportation Research Interdisciplinary Perspectives, 3, 100071.

<https://doi.org/10.1016/j.trip.2019.100071>

Thoutam, V. (2021). *Physical Design, Origins and Applications of IOT. 01.*

Virgen-Cobos, G. H., Gómez-Díaz, J. D., Borja-de La Rosa, Ma. A., Ramírez

García, A. G., & Monterroso Rivas, A. I. (2022). Sistema de monitoreo meteorológico en un bosque templado a partir de análisis del paisaje.

Revista Mexicana de Ciencias Forestales, 13(74), 145-173.

<https://doi.org/10.29298/rmcf.v13i74.1260>

Zeadally, S., Shaikh, F. K., Talpur, A., & Sheng, Q. Z. (2020). Design architectures

for energy harvesting in the Internet of Things. *Renewable and Sustainable*

Energy Reviews, 128, 109901. <https://doi.org/10.1016/j.rser.2020.109901>

ANEXOS

ANEXO A

Codigo implementacion ESP8266

```
#include <Ubidots.h>
```

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

```
#include <SPI.h>
```

```
#include <Adafruit_Sensor.h>
```

```
#include "Adafruit_BME680.h"
```

```
#define SEALEVELPRESSURE_HPA (1013.25)
```

```
const char* UBIDOTS_TOKEN = "BBUS-
```

```
WADsfMrhkaSAVeulSujTkcZ04WPvbT"; // Put here your Ubidots TOKEN
```

```
const char* WIFI_SSID = "REDPON 456"; // Put here your Wi-Fi SSID
```

```
const char* WIFI_PASS = "kwvuW4np75"; // Put here your Wi-Fi password
```

```

const int analogInPin = A0; // ESP8266 Analog Pin ADC0 = A0
const int sensorMaxValue = 1024; // Valor máximo del sensor
const float degreesPerUnit = 360 / sensorMaxValue; // Grados por cada unidad del
sensor
int sensorValue = 0;

const byte interruptPin = 14; // Or other pins that support an interrupt
unsigned int Debounce_Timer, Current_Event_Time, Last_Event_Time,
Event_Counter;
float WindSpeed;

Ubidots ubidots(UBIDOTS_TOKEN, UBI_HTTP);
Adafruit_BME680 bme; // I2C

void setup() {
  Serial.begin(115200);
  ubidots.wifiConnect(WIFI_SSID, WIFI_PASS);
  while (!Serial);
  Serial.println(F("BME680 async test"));

  if (!bme.begin()) {
    Serial.println(F("Could not find a valid BME680 sensor, check wiring!"));
    while (1);
  }
  // Set up oversampling and filter initialization
  bme.setTemperatureOversampling(BME680_OS_8X);
  bme.setHumidityOversampling(BME680_OS_2X);
  bme.setPressureOversampling(BME680_OS_4X);
  bme.setIIRFilterSize(BME680_FILTER_SIZE_3);
  bme.setGasHeater(320, 150); // 320°C for 150 ms
  ////////////////////////////////////WINDSPEED INTERRUPTS////////////////////////////////
  pinMode(interruptPin, INPUT);
  noInterrupts(); // Disable interrupts during set-up
  attachInterrupt(digitalPinToInterrupt(interruptPin), WSpeed_ISR, FALLING);
  //Respond to a LOW on the interrupt pin and goto WSpeed_ISR
  timer0_isr_init(); // Initialise Timer-0
  timer0_attachInterrupt(Timer_ISR); // Goto the Timer_ISR function when
an interrupt occurs
  timer0_write(ESP.getCycleCount() + 800000000L); // Pre-load Timer-0 with a time
value of 10-second
  interrupts(); // Enable interrupts
}

```

```

void loop() {
  //////////////////////////////////////
  sensorValue = analogRead(analogInPin);
  float degrees = sensorValue * 0.3515625;

  if (degrees < 0) {
    degrees += 360.0;
  }

  //////////////////////////////////////
  unsigned long endTime = bme.beginReading();
  if (endTime == 0) {
    Serial.println(F("Failed to begin reading :("));
    return;
  }

  delay(50); // This represents parallel work.

  if (!bme.endReading()) {
    Serial.println(F("Failed to complete reading :("));
    return;
  }

  delay(10000);

  ubidots.add("bme680_Temperature", bme.temperature); // Change for your
variable name
  ubidots.add("bme680_Pressure", (bme.pressure / 100.0)); // Change for your
variable name
  ubidots.add("bme680_Humidity", bme.humidity); // Change for your variable name
  ubidots.add("bme680_Gas", (bme.gas_resistance / 1000.0)); // Change for your
variable name
  ubidots.add("bme680_Altitude",
bme.readAltitude(SEALEVELPRESSURE_HPA)); // Change for your variable name
  ubidots.add("vantage_Pro2_Wind_Direction", degrees); // Change for your
variable name
  ubidots.add("vantage_Pro2_Wind_Speed", WindSpeed);

  bool bufferSent = false;
  bufferSent = ubidots.send(); // Will send data to a device label that matches the
device Id

  if (bufferSent) {

```

```

// Do something if values were sent properly
Serial.println("Values sent by the device\n");
Serial.println("*****");
}
WiFi.disconnect(true); // Desconectar Wi-Fi antes de dormir profundamente
ESP.deepSleep(600e6); // 600 segundos * 1,000,000 (microsegundos por
segundo)

}

// Interrupt service routine
ICACHE_RAM_ATTR void WSpeed_ISR (void) {
  if (!(millis() - Debounce_Timer) < 5) {
    Debounce_Timer = millis(); // Set debouncer to prevent
false triggering
    Event_Counter++;
  }
}

void Timer_ISR (void) { // Timer reached zero,
now re-load it to repeat
  timer0_write(ESP.getCycleCount() + 800000000L); // Reset the
timer, do this first for timing accuracy
  WindSpeed = Event_Counter*(2.25/10.0); // V = P(2.25/T)
  //Serial.println(Event_Counter);
  Event_Counter =0;
}

```

ANEXO B

Codigo HTML

```

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Estación Meteorológica V 1.0</title>
<link rel="stylesheet" href="estilos.css"> <!-- Enlace al archivo CSS externo -->
</head>
<body>

<div class="dashboard">
  <div class="item full-width">ESTACION METEOROLOGICA V 1.0</div>
  <div class="item half-width">

```

UBICACIÓN

Comunidad: San Pedro Ahuacatlan

Municipio: San Juan del Rio

Estado: Queretaro

Coordenadas: 20.42949164558877, -99.98406872140148

</div>

<div class="item" style="grid-column-end: span 3;">

<iframe width="100%" height="932" frameborder="0"

src="https://stem.ubidots.com/app/dashboards/public/widget/7-z8HzEjdvTR-czNmX49a1zVWcTMII-YKND7iLrrhrM?embed=true"></iframe>

</div>

<!-- Primer renglón de iframes -->

<div class="item">

<iframe width="100%" height="932" frameborder="0"

src="https://stem.ubidots.com/app/dashboards/public/widget/j5kmoaOcO2liNxTszx7oE8-oNj1EvnCthO3U2p5cZ_Q?embed=true"></iframe>

</div>

<div class="item">

<iframe width="100%" height="932" frameborder="0"

src="https://stem.ubidots.com/app/dashboards/public/widget/VzPd5iM5Zcl90QkNejrbWgoOTvk5W5qGyE42ApneZBQ?embed=true"></iframe>

</div>

<div class="item">

<iframe width="100%" height="932" frameborder="0"

src="https://stem.ubidots.com/app/dashboards/public/widget/FKegkrBCQEmlgGji64mslmAVLqxfKxgH0D9fNPsu4NU?embed=true"></iframe>

</div>

<div class="item">

<iframe width="100%" height="932" frameborder="0"

src="https://stem.ubidots.com/app/dashboards/public/widget/I7JL0OYa_BsnFGt6V95OaqPVZYMqKg-KIA-FM2eVnRE?embed=true"></iframe>

</div>

<div class="item">

<iframe width="100%" height="932" frameborder="0"

src="https://stem.ubidots.com/app/dashboards/public/widget/gCg-tXzgGv_FNO1JTFAo-VqPxTtcJaWcMmF_Ts0vUwM?embed=true"></iframe>

</div>

<!-- Segundo renglón de iframes -->

<div class="item">

```

    <iframe width="100%" height="932" frameborder="0"
src="https://stem.ubidots.com/app/dashboards/public/widget/0ZGoeL-Y_tFoY66xB-
uL9bZBY77femnzCwgManOeQml?embed=true"></iframe>
</div>
<div class="item">
    <iframe width="100%" height="932" frameborder="0"
src="https://stem.ubidots.com/app/dashboards/public/widget/n9fZuxKn2r9j0ziwiu22
3niHuqbmB9BbAn1LnPqQd4w?embed=true"></iframe> </div>
<div class="item">
    <iframe width="100%" height="932" frameborder="0"
src="https://stem.ubidots.com/app/dashboards/public/widget/FKegkrBCQEmlgGji6
4mslmAVLqxfKxgH0D9fNPsu4NU?embed=true"></iframe> </div>
<div class="item">
    <iframe width="100%" height="932" frameborder="0"
src="https://stem.ubidots.com/app/dashboards/public/widget/67CETHf7USkZNDv_
yfo9_OZatouza-ul4Nb-PH869HY?embed=true"></iframe>
</div>
<div class="item">
    <iframe width="100%" height="932" frameborder="0"
src="https://stem.ubidots.com/app/dashboards/public/widget/2N1MUvGkhHwboyG
zAheoSteaZ-2XA9oElfX-_VzZ8jU?embed=true"></iframe>
</div>
</div>

</body>
</html>

```

ANEXO C

Código CSS

```

/* Estilos para el dashboard general */
body {
    font-family: 'Arial', sans-serif;
    background-color: #eaeaea; /* Fondo general más claro */
    color: #333; /* Color de texto general */
}
.dashboard {
    display: grid;
    grid-template-columns: repeat(5, 1fr); /* 5 columnas iguales */
    grid-template-rows: auto auto auto; /* Título, y dos filas de contenido */
    gap: 10px;
    padding: 10px;
}

```

```
margin: auto; /* Centrar la dashboard en la página */
max-width: 1200px; /* Ancho máximo */
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); /* Sombra alrededor */
}
```

```
/* Estilos para los elementos individuales */
.item {
border: 1px solid #000;
padding: 5px;
display: flex;
justify-content: center;
align-items: center;
}
```

```
/* Estilos para el título principal */
.full-width {
grid-column: 1 / -1; /* Abarca todas las columnas */
font-size: 24px; /* Tamaño de letra más grande */
background-color: #4CAF50; /* Fondo verde */
color: white; /* Letra blanca */
font-weight: bold; /* Letra en negrita */
text-align: center; /* Centrar texto */
}
```

```
/* Estilos para la sección de ubicación */
.half-width h2 {
background-color: #2196F3; /* Fondo azul para el título de la sección */
color: white; /* Letra blanca */
padding: 5px;
margin-top: 0; /* Eliminar el margen superior */
}
.half-width ul {
background-color: #f2f2f2; /* Fondo gris claro */
padding: 10px;
border-radius: 5px; /* Bordes redondeados */
}
```

```
/* Estilos para los iframes */
.item iframe {
width: 100%;
height: 100%;
}
```

```
max-width: 100%;
height: auto; /* Altura automática para mantener la proporción */
border: none; /* Eliminar el borde del iframe */
}
```

ANEXO D

Código análisis climatológico

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
# Cargar y limpiar los datos de San Pedro Ahuacatlan
data_san_pedro = pd.read_csv('SanPedroAhuacatlan.csv', encoding='ISO-8859-1')
data_san_pedro_cleaned = data_san_pedro.drop(index=range(8))
data_san_pedro_cleaned.columns = ['timestamp', 'temperature', 'pressure', 'humidity']
data_san_pedro_cleaned = data_san_pedro_cleaned.iloc[1:]
data_san_pedro_cleaned['timestamp'] =
pd.to_datetime(data_san_pedro_cleaned['timestamp'], format='mixed')
data_san_pedro_cleaned['temperature'] =
pd.to_numeric(data_san_pedro_cleaned['temperature'], errors='coerce')
data_san_pedro_cleaned['pressure'] =
pd.to_numeric(data_san_pedro_cleaned['pressure'], errors='coerce')
data_san_pedro_cleaned['humidity'] =
pd.to_numeric(data_san_pedro_cleaned['humidity'], errors='coerce')
# Cargar y limpiar los datos de las otras estaciones
def clean_station_data(data):
    data_cleaned = data.drop(index=range(8))
    data_cleaned.columns = ['timestamp', 'temperature', 'humidity', 'pressure']
    data_cleaned['timestamp'] = pd.to_datetime(data_cleaned['timestamp'],
format='%d/%m/%Y %H:%M', errors='coerce')
    data_cleaned['temperature'] = pd.to_numeric(data_cleaned['temperature'],
errors='coerce')
    data_cleaned['pressure'] = pd.to_numeric(data_cleaned['pressure'],
errors='coerce')
    data_cleaned['humidity'] = pd.to_numeric(data_cleaned['humidity'],
errors='coerce')
    return data_cleaned
data_huichapan = pd.read_csv('Estacion_HUICHAPANSMN.csv', encoding='ISO-8859-1')
data_huimilpan = pd.read_csv('Estacion_HUIMILPAN.csv', encoding='ISO-8859-1')
data_huichapan_cleaned = clean_station_data(data_huichapan)
data_huimilpan_cleaned = clean_station_data(data_huimilpan)
```

```

# Agregar información de ubicación
data_san_pedro_cleaned['latitud'] = 20.42981442
data_san_pedro_cleaned['longitud'] = -99.98411821
data_san_pedro_cleaned['altitud'] = 1920
data_huichapan_cleaned['latitud'] = 20.38861111
data_huichapan_cleaned['longitud'] = -99.66388889
data_huichapan_cleaned['altitud'] = 2080
data_huimilpan_cleaned['latitud'] = 20.39027778
data_huimilpan_cleaned['longitud'] = -100.2836111
data_huimilpan_cleaned['altitud'] = 2280
# Combinar los datos para análisis comparativo
data_san_pedro_cleaned['station'] = 'San Pedro Ahuacatlan'
data_huichapan_cleaned['station'] = 'Huichapan'
data_huimilpan_cleaned['station'] = 'Huimilpan'
combined_data = pd.concat([data_san_pedro_cleaned, data_huichapan_cleaned,
data_huimilpan_cleaned], ignore_index=True)
combined_data = combined_data.dropna()
# Visualizar distribuciones
fig, axes = plt.subplots(nrows=3, ncols=1, figsize=(10, 15))
sns.boxplot(x='temperature', y='station', data=combined_data, ax=axes[0])
sns.boxplot(x='pressure', y='station', data=combined_data, ax=axes[1])
sns.boxplot(x='humidity', y='station', data=combined_data, ax=axes[2])
plt.tight_layout()
plt.show()
# Análisis de fiabilidad
consistency_summary = data_san_pedro_cleaned[['temperature', 'pressure',
'humidity']].describe()
Q1 = data_san_pedro_cleaned[['temperature', 'pressure', 'humidity']].quantile(0.25)
Q3 = data_san_pedro_cleaned[['temperature', 'pressure', 'humidity']].quantile(0.75)
IQR = Q3 - Q1
outliers = ((data_san_pedro_cleaned[['temperature', 'pressure', 'humidity']] < (Q1 -
1.5 * IQR)) |
            (data_san_pedro_cleaned[['temperature', 'pressure', 'humidity']] > (Q3 + 1.5
* IQR))).sum()
comparison_summary = combined_data.groupby('station')[['temperature',
'pressure', 'humidity']].mean()
time_diffs = data_san_pedro_cleaned['timestamp'].diff().dropna()
irregular_intervals = time_diffs.describe()
# Combinando los datos de las tres estaciones
combined_data_updated = pd.concat([data_san_pedro_cleaned,
data_huichapan_cleaned, data_huimilpan_cleaned], ignore_index=True)
# Convertir todas las marcas de tiempo a tz-naive
combined_data_naive = combined_data_updated.copy()

```

```
combined_data_naive['timestamp'] =
combined_data_naive['timestamp'].apply(lambda x: x.replace(tzinfo=None))
# Encontrar el punto de inicio común más temprano entre las tres estaciones
start_date_naive = combined_data_naive['timestamp'].min()
# Filtrar los datos para incluir solo mediciones desde el punto de inicio común
filtered_data_aligned = combined_data_naive[combined_data_naive['timestamp']
>= start_date_naive]
# Crear la gráfica con el mismo punto de inicio
plt.figure(figsize=(15, 7))
# Graficar la temperatura para cada estación con el mismo punto de inicio
for station, group_data in filtered_data_aligned.groupby('station'):
    plt.plot(group_data['timestamp'], group_data['temperature'], label=station)
plt.title('Temperatura a lo largo del tiempo por Estación (Alineado)')
plt.xlabel('Fecha y Hora')
plt.ylabel('Temperatura (°C)')
plt.legend()
plt.xticks(rotation=45)
plt.grid(True)
plt.tight_layout()
plt.show()
```