



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Politécnica Superior de Gandia

Creación de un Observatorio Marino Basado en una Red
de Sensores para la Monitorización de las Variables de
Zonas Marítimo-Costeras

Trabajo Fin de Grado

Grado en Tecnologías Interactivas

AUTOR/A: Zaragoza Esquerdo, Miguel

Tutor/a: Sendra Compte, Sandra

Cotutor/a: Lloret Mauri, Jaime

Cotutor/a: Parra Boronat, Lorena

CURSO ACADÉMICO: 2023/2024



Agradecimientos

Me gustaría agradecer este trabajo de fin de grado a mis tutores Sandra Sendra Compte, Lorena Parra Boronat y Jaime Lloret Mauri por la ayuda proporcionada en el proyecto. Por otra parte, me gustaría hacer una especial mención a mi familia en concreto a mi padre por ayudarme y apoyarme en cada uno de mis proyectos y andaduras durante toda la vida. Y por último me gustaría agradecerlo también a mi pareja Carmen y a mi amigo Jaume por acompañarme siempre en todos los procesos que vivo y ser un pilar clave en mi vida.



Resumen

En las zonas costeras, el medio marino es una de las principales fuentes de ingresos tanto por la pesca como por las actividades ligadas al turismo. Es por ello, que su conservación y monitorización son importantes y las nuevas tecnologías nos ofrece un sistema de alerta temprana para detectar la contaminación y otras situaciones anómalas. El objetivo de este trabajo de fin de grado es desarrollar e implementar la tecnología necesaria para un observatorio marítimo, que se integrará en un proyecto de investigación que monitorice las condiciones del océano durante un largo periodo de tiempo para poder hacer investigaciones medioambientales en distintos emplazamientos de la costa. El observatorio llevará una serie de nodos multiparamétricos y cámaras a lo largo de la costa. En este trabajo desarrollaremos uno de esos nodos que llevarán incorporado un sensor de conductividad, una sonda de temperatura y unos sensores de turbidez marina que miden las partículas de suspensión en el agua con foto emisores de luz infrarroja y luz visible. Los sensores de turbidez y conductividad son desarrollos propios mientras que el sensor de temperatura es un sensor comercial. Se incorporará un sistema de comunicación ofreciendo dos alternativas, un sistema de transmisión inalámbrica como puede ser Lora y un sistema de transmisión por cable Ethernet. Concretamente, en la tecnología Lora modificaremos la configuración y los buffers para poder transmitir el video de las cámaras. Estas tecnologías se utilizarán para llevar la información censada al servidor instalada en la base donde este instalada el ordenador para monitorización. El servidor estará implementado en el puerto destino que constará de un servidor TCP para la transmisión por cable o, en el caso de que sea inalámbrica, un nodo receptor. Estos sistemas trasladarán la información a un servidor Apache que está conectado con un servidor Express que tramitará las peticiones HTTP que emita el nodo multiparamétrico. La información se guardará en una base datos MySQL y se podrá visualizar desde una aplicación web montada en REACT JS.

Palabras clave

Lora (Largo Alcance), Internet de las Cosas (IoT), Calidad del Agua, Monitorización Ambiental, Servidor REST, Base de Datos, Nodo Sensor.

Abstract

In coastal areas, the marine environment is one of the main income sources from fishing and activities linked to tourism. Therefore, its conservation and monitoring are important and new technologies offer us an early warning system to detect pollution and other anomalous situations. The objective of this final degree project is to develop and implement the necessary technology for a maritime observatory, which will be integrated into a research project that monitors the conditions of the ocean over a long period of time to be able to carry out environmental research in different locations in the coast. The observatory will consist of a series of multiparametric nodes and cameras along the coast. In this work, we will develop one of those nodes. The node will incorporate a conductivity sensor, a temperature probe, and some marine turbidity sensors that measure suspended particles in the water using infrared light and visible light photoemitters. Turbidity and conductivity sensors are our own developments, while temperature sensors are commercial sensors. A communication system offering two alternatives, a wireless transmission system such as LoRa and an Ethernet cable transmission system, will be incorporated. Specifically, in LoRa technology, we will modify the configuration and buffers to transmit the video from the cameras. These technologies will be used to take the sensed information to the server installed in the base where the computer is installed for monitoring. The server will be implemented in the destination port, which will consist of a TCP server for cable transmission or, in the case of wireless transmission, a receiving node. These systems will transfer the information to an Apache server that is connected to an Express server that will process the HTTP requests issued by the multiparameter node. The information will be saved in a MySQL database and can be viewed from a web application mounted in REACT JS.

Key words

LoRa (Long Range), Internet of Things (IoT), Water Quality, Environmental Monitoring, REST Server, Database, Sensor Node.

Resum

En les zones costaneres, el medi marí és una de les principals fonts d'ingressos tant per la pesca com per les activitats vinculades al turisme. És per això que la seua conservació i monitorització són importants i les noves tecnologies ens ofereixen un sistema d'alerta primerenca per a detectar la contaminació i altres situacions anòmales. L'objectiu d'aquest treball de fi de grau és desenvolupar i implementar la tecnologia necessària per a un observatori marítim, que s'integrarà en un projecte de recerca que monitoritze les condicions de l'oceà durant un llarg període de temps per a poder realitzar investigacions mediambientals en diferents emplaçaments de la costa. L'observatori portarà una sèrie de nodes multiparamètrics i càmeres al llarg de la costa. En aquest treball desenvoluparem un d'aquests nodes que portaran incorporat un sensor de conductivitat, una sonda de temperatura i uns sensors de turbidesa marina que mesuren les partícules de suspensió en l'aigua amb fotoemissors de llum infraroja i llum visible. Els sensors de turbidesa i conductivitat són desenvolupaments propis mentre que el sensor de temperatura és un sensor comercial. S'incorporarà un sistema de comunicació oferint dues alternatives, un sistema de transmissió sense fil com pot ser LoRa i un sistema de transmissió per cable Ethernet. Concretament, en la tecnologia LoRa modificarem la configuració i els buffers per a poder transmetre el vídeo de les càmeres. Aquestes tecnologies s'utilitzaran per a portar la informació sensorial al servidor instal·lat en la base on estiga instal·lat l'ordinador per a la monitorització. El servidor estarà implementat en el port destí que constarà d'un servidor TCP per a la transmissió per cable o, en el cas que siga sense fil, un node receptor. Aquests sistemes traslladaran la informació a un servidor Apache que està connectat amb un servidor Express que tramitarà les peticions HTTP que emeta el node multiparamètric. La informació es guardarà en una base de dades MySQL i es podrà visualitzar des d'una aplicació web muntada en REACT JS.

Paraules clau

Lora (Llarg Alcance), Internet de les Coses (IoT), Qualitat de l'Aigua, Monitorització Ambiental, Servidor REST, Base de Dades, Node Sensor.



Índice

Índice de Tablas	1
Índice de Figuras	2
Capítulo 1. Introducción.....	4
1.1 Introducción	4
1.2 Precedentes en la Escuela Politécnica Superior de Gandia	5
1.3 Objetivos	5
1.4 Diagrama de Gantt	6
1.5 Estructura del proyecto.....	6
Capítulo 2. Estado del arte	7
2.1 Sensores.....	7
2.2 Comunicaciones Lora.....	9
Capítulo 3. Propuesta de nodo sensor multiparamétrico	11
3.1 Sensores.....	11
3.1.1 Sensor de conductividad.....	11
3.1.2 Sensor de temperatura	12
3.1.3 Sensor de turbidez	13
3.2 Microcontroladores y conexiones	14
3.2.1 Microcontrolador.....	14
3.2.2 Esquema de conexiones	14
3.5 Encapsulado	15
3.6 Transmisión por cable	15
3.7 Transmisión inalámbrica	17
3.7.1 Wi-Fi (IEEE 802.11) para comunicación en tierra.....	17
3.7.2 LORA	18
3.8 Servidor y base de datos.....	21
3.8.1 Servidor montado en Node Express	21
3.8.2 Base de datos.....	24
3.9 Cliente web.....	25
3.9.1 Tabla principal.....	26
3.9.2 Gráfica general	26
3.9.3 Exportación de los datos a formato Excel	27
3.9.4 Informes	27
Capítulo 4. Desarrollo	28



4.1 Tubo para obtener las medidas del agua del mar	28
4.2 PCB sensor de conductividad.....	29
4.3 Adaptación de la bobina de conductividad al tubo.....	29
4.4 Abrazadera para el sensor de IR de turbidez	30
4.5 Adaptación del sensor RGB de turbidez	31
Capítulo 5. Pruebas	32
5.1 Calibrado de sensores.....	32
5.1.1 Calibrado del sensor de conductividad.....	32
5.1.2 Calibrado del sensor de temperatura	36
5.1.3 Calibrado del sensor de turbidez	36
5.2 Medidas de vídeo en Lora	39
5.2.1 Tamaño de bloque	39
5.2.2 Evaluación de la calidad máxima del video transmitido	39
5.2.3 Evaluación de MAT para diferentes tipos de extensiones de video	41
5.2.4 Desempeño del sistema propuesto en interiores.....	41
5.2.5 Desempeño del sistema propuesto en exteriores.....	42
5.3 Configuración de los nodos para la transmisión Lora	43
5.3.1 Medidas de distancia en Lora.....	43
Capítulo 6. Conclusiones.....	45
6.1 Cumplimiento del objetivo.....	45
6.2 Conclusiones sobre el proyecto.....	45
6.3 Problemas encontrados y cómo se han solucionado	46
6.4 Aportaciones personales.....	46
6.5 Publicaciones.....	46
6.6 Futuras líneas de trabajo.....	47
Capítulo 7. Referencias	48



Índice de Tablas

Tabla 1 Diagrama de Gantt.....	6
Tabla 2 Configuración de los módulos Lora	18
Tabla 3 Muestras de calibración.....	32
Tabla 4 Conductividad de muestras de calibración y validación	33
Tabla 5 Tabla de los diferentes de coeficientes de correlación por modelo.....	34
Tabla 6 Resultados de la comparación de modelos alternativos	35

Índice de Figuras

Figura 1 Diagrama electrónico del sensor de conductividad.....	11
Figura 2 Sensor de conductividad montado en protoboard.....	12
Figura 3 Código para medir la conductividad en Arduino	12
Figura 4 Sonda de temperatura DS18B20.....	13
Figura 5 Ejemplo de código el sensor de temperatura DS18B20.....	13
Figura 6 Esquema de conexiones empleado en el proyecto	15
Figura 7 Ejemplo de código de la transmisión de los datos por cable	17
Figura 8 Ejemplo de código para transmitir los datos por WIFI.....	18
Figura 9 Nodo transmisor Lora montado	19
Figura 10 Pseudocódigo de la transmisión por Lora del emisor.	20
Figura 11 Pseudocódigo de la transmisión por Lora del receptor.....	21
Figura 12 Código del archivo server.js	22
Figura 13 Código del archivo sensorRoute.js	22
Figura 14 Código del archivo sensorController.js.....	23
Figura 15 Código del archivo sensorService.js.....	24
Figura 16 Diseño de la base de datos	25
Figura 17 UX tabla principal de la web	26
Figura 18 UX gráfica general de la web.....	26
Figura 19 UX de la exportación de los datos en Excel de la web	27
Figura 20 UX de la zona de informes de la web	27
Figura 21 Disposición de los elementos del observatorio en la demo	28
Figura 22 Prototipo del tubo donde se toman las medidas.....	29
Figura 23 Circuito impreso del sensor de conductividad	29
Figura 24 Bobina primaria y secundaria integrada al tubo	30
Figura 25 Abrazadera para el sensor de IR de turbidez	30
Figura 26 Adaptación del sensor RGB de turbidez.....	31
Figura 27 Análisis descriptivo para las muestras de la calibración.....	34
Figura 28 Modelo inversa de Y para salinidad.....	34
Figura 29 Modelo de regresión lineal para salinidad	35
Figura 30 Comparación de la sonda de laboratorio respecto a la sonda DS18B20.....	36
Figura 31 Análisis descriptivo de los datos.....	37
Figura 32 Representación de los grupos de turbidez.....	37
Figura 33 Recta de regresión de la materia vegetal en descomposición	38
Figura 34 Recta de regresión de la materia vegetal fresca	38
Figura 35 Gráfica MAT respecto al tamaño de bloque	39



Figura 36 MAT en diferentes tamaños de video	40
Figura 37 Variación del MAT en diferentes calidades de video	40
Figura 38 Variación del MAT para distintos tamaños de video.....	41
Figura 39 Efecto del tamaño de bloque al MAT	42
Figura 40 RTT respecto a la calidad del video a distancia.....	43
Figura 41 Visual de donde se envió el video en exteriores	43
Figura 42 RTT haciendo un ping en Lora	44
Figura 43 Mapa de las localizaciones y distancias donde se han hecho las medidas.....	44



Capítulo 1. Introducción

1.1. Introducción

Hoy en día la situación demográfica en España sitúa a la mayoría de la población en la costa [1]. Todo ello lleva una serie de complicaciones a la hora de llevar a cabo la preservación de los ecosistemas marinos costeros. Por lo tanto, proviene de que una masificación en la costa provoca un aumento de zonas críticas donde puede crearse una alteración del medio marino. Esto puede ser por ejemplo en piscifactorías, zonas de playa donde se realizan actividades acuáticas, zonas de desagüe al océano o en puertos marítimo-pesqueros. Por ello es muy importante llevar a cabo una detección y monitorización de posibles cambios en las variables costeras del océano para su posterior subsanación.

Como se necesita muchas sondas para monitorizar dichas zonas aumenta mucho el coste de esta infraestructura. De ello nace la necesidad de crear sensores de bajo coste que abaraten los costes y den un buen servicio en los nodos multiparamétricos. Los sensores deberán de parametrizar el estado del ecosistema marino como puede ser la temperatura, salinidad, conductividad, turbidez o el pH. Como por ejemplo el uso de sensores de conductividad marina [2] de bajo coste o el uso de sensores de pH del medio en zonas costeras [3]. Otro caso que requiere de atención es el uso de video en tiempo real con el uso de cámaras con sistemas de comunicaciones Lora para la visión del espacio que se está estudiando.

Al estar los sensores obteniendo datos en zonas remotas se requiere de una infraestructura de comunicaciones para enviar los datos a los servidores para la captura de datos y su recopilación [4]. Para ello se requiere de una arquitectura de IoT para el envío y recepción de los datos [5]. Los sistemas por ende deberán estar enviando datos por el medio acuoso y pueden ser de dos tipos inalámbricos o por cable.

Los medios inalámbricos como puede ser IEEE 802.11 se han hecho estudios de la viabilidad de enviar datos debajo del agua como se puede ver en [6] [7]. Otra alternativa es el envío de datos por transmisión punto a punto Lora [8] que al tener menor frecuencia de transmisión puede dar una mejor cobertura en la señal. Por otra parte, una transmisión por cable puede dar problemas por la distancia, costo y implementación de la línea de transmisión del nodo al servidor del mismo [9]. Otro aliciente a tener en cuenta es la escalabilidad de la red de sensores para la colaboración de las mismas tal como se puede ver en [10] donde dependiendo de la topología de red que montemos podemos abaratar y mejorar la comunicación en ella.

En este proyecto se hace un estudio para crear una alternativa de bajo coste para poder monitorizar y observar el medio marino. Para ello se investigan y desarrollan sensores de creación propia para dar cobertura a la observación como el uso de sensores comerciales como sondas de temperatura. Para ver que alternativa de comunicación del nodo al servidor da mejores prestaciones se hace un estudio de las alternativas de comunicación como puede ser Wi-Fi, Lora o transmisión por cable puede dar mejores prestaciones. También se implementa un servidor con un cliente web para que los usuarios puedan trabajar con los datos obtenidos. Y una vez visto todos los aspectos se monta el observatorio marino para su puesta en escena.



1.2 Precedentes en la Escuela Politécnica Superior de Gandia

En esta sección vamos a ver proyectos de final de grado y líneas de investigación pasadas que se han hecho en la universidad politécnica de Valencia.

En 2022 María Soto Pérez pretendió diseñar y desarrollar un sistema de monitorización de la calidad del agua de las zonas costeras. En concreto, se centró en la detección de contaminantes presentes en el agua procedente de los cultivos cercanos y que, a pesar de no ser tratada correctamente, acaba desembocando en el mar. Además de ello, el diseño un dispositivo tendría otras funcionalidades como la de medir la temperatura del agua, el nivel de turbidez y las mareas o corrientes del mar. Todo ello, sería mostrado en una aplicación web disponible para todo aquel que desee ver una información sobre la costa en tiempo real. Para llevar a cabo dicha tarea, se hizo uso de la tecnología de Lora WAN [11].

En 2022 Gómez Fernández-Pacheco, Javier desarrolló de un sistema de sensores para la monitorización de Posidonia oceánica para determinar aquellos parámetros que pueden ser utilizados para monitorizar la calidad del agua con el fin de conservar las praderas. Que la monitorización de estos parámetros se realice con sistemas de bajo coste, a fin de poder utilizar una gran cantidad de sensores en el medio. Se determinaron 2 parámetros que cumplen dichos requisitos: la turbidez y la conductividad. En el caso de la conductividad se ha optado por un sensor de tipo inductivo y en la turbidez por uno de óptico, Por un lado, no obtuvimos resultados satisfactorios con el sensor inductivo, pues las diferencias de lectura entre las muestras de 25 mg/l y 35mg/l eran mínimas. Con el sensor de turbidez se consiguió una diferencia de voltaje de 0,615 V, con un error de calibración de 0,35 NTU y de verificación de 0,25 [12].

1.3 Objetivos

El objetivo del proyecto es crear una sonda multiparamétrica para un proyecto de la Generalitat Valenciana enmarcado dentro de un Proyecto Europeo de investigación que monitorice las condiciones del océano. Durante el periodo de un año para poder hacer investigaciones medioambientales en distintos emplazamientos de la costa. Como pueden ser puertos pesqueros, petroleras, piscifactorías, playas, zonas de vertidos o lugares de arrastre de la corriente. Para ello debe de dar cabida un instrumental que pueda operar en esas condiciones y almacenar la información para posteriormente enviarla a tierra para poder ser procesada. Otro objetivo es la realización de investigaciones de las tecnologías necesarias para poder llevarla a cabo como puede ser de sistemas de transmisiones, sistemas de control y procesamiento, sistemas de interconexión o sistemas de recepción y almacenamiento. Por último, un pilar importante es desarrollar sensores de bajo coste para la monitorización de los datos en el océano. Una vez realizada esta investigación se llevará a cabo la puesta en escena de este proyecto. De ellos se desprenden los siguientes subobjetivos:

1. Desarrollar, calibrar y aislar sensores de bajo coste para conductividad, turbidez y temperatura.
2. Proponer y testear un sistema de comunicación basado en Lora, Ethernet y Wi-Fi que permita el envío de vídeo y datos de los sensores.
3. Diseño de una base de datos para acceso a los datos tomados por el nodo.

4. Realizar diseños electrónicos para adaptar la señal para que pueda ser recogida adecuadamente por el Arduino.

1.4 Diagrama de Gantt y metodología

En esta sección, veremos el diagrama de Gantt utilizado durante la producción del proyecto viendo en él las distintas fases en la que se ha organizado el trabajo.

Tabla 1 Diagrama de Gantt



Como vemos en la Tabla 2, se ha dividido el proyecto en 8 tareas. En la primera se trata en febrero el tema de la planificación y diseño de las arquitecturas y las tecnologías utilizadas en el proyecto. Posteriormente se lleva a cabo la elaboración de las tecnologías que engloban a las comunicaciones, sensores y el cliente-servidor. Una vez desarrolladas estas tecnologías se lleva la implementación de la demo para hacer la prueba del observatorio. Una vez implementado nos centramos en el testeo y las pruebas de la demo. Por último, se hace un seguimiento de los datos obtenidos en la demo y la creación de la memoria de todos los temas tratados en el proyecto.

La metodología utilizada a lo largo del proyecto es CDIO, donde la aplicamos de la siguiente manera. Primero, se concibe la necesidad de las tareas. Segundo, diseña la arquitectura de las tareas. Tercero, se implementan las tareas en base al diseño. Por último, se opera y se hacen pruebas para evaluar el rendimiento de la funcionalidad nueva.

1.5 Estructura del proyecto

La estructura del proyecto cuenta de 6 partes. En la sección 2 se hace un estudio del estado del arte en lo que respecta al tema de los sensores utilizados y en la segunda parte en lo que respecta a sistemas de transmisión inalámbrica como puede ser Lora. En la sección 3 se hace una propuesta detallada de los sensores que se van a utilizar, el microcontrolador utilizado y conexiones, el encapsulado que se va a utilizar, la propuesta de los nodos Lora utilizados para las pruebas y por último la parte del servidor y bases de datos como la parte del cliente para que el usuario monitorice los datos. En la sección 4 se desarrollarla todos los pasos empleados para el desarrollo del observatorio marino. En la sección 5 se detallarán todas las pruebas empleadas para el calibrado y el testeo de las tecnologías empleadas en el proyecto. Y por último en la sección 6 daré las conclusiones y líneas futuras de investigación del proyecto.

Capítulo 2. Estado del arte

En esta sección se va a tratar el estado del arte en el ámbito de los sensores y en el de las comunicaciones en Lora.

2.1 Sensores

En este apartado se muestran diferentes estudios sobre la implementación de sensores para determinar la calidad del agua (conductividad y turbidez) en ambientes acuáticos. Además, se presentan estudios relacionados con la teledetección y cómo se han interconectado diferentes técnicas para establecer valores de conductividad y turbidez para la toma de decisiones.

Brinda Das y P.C. Jain, en 2017 [13], implementaron un sistema de calidad del agua en tiempo real utilizando diferentes sensores (pH, conductividad, temperatura). Utilizaron el módulo ZigBee para transferir los datos obtenidos al microcontrolador. El módulo GSM envía estos datos a un teléfono inteligente o computadora. Además, el sistema envía mensajes para alertarles de vertidos que puedan contaminar el agua. En 2015, Parra et al. [14] desarrollaron un sensor de bajo coste basado en la conductividad. Para ello, utilizaron dos bobinas utilizando el método de inductancia mutua. Probaron cinco prototipos y concluyeron que las bobinas que daban mejores resultados eran las de 40 vueltas y 80 vueltas (alimentadas e inducidas, respectivamente). Con los resultados obtenidos, extrajeron una ecuación que permitió calcular la conductividad del agua con alta precisión. Wang et al. en 2020 [15] revisaron los sistemas de conductividad. Los sensores utilizados se basaron en polímeros conductores (CP). Las PC están compuestas principalmente por polianilina, polipirrol y poli (3,4-etileno dioxitiofeno). Estos se prepararon en diferentes condiciones de polimerización y se utilizaron como diversos sensores. Se observó que el CP y otros materiales de detección, como metales, óxidos metálicos, etc., muestran un alto rendimiento de detección.

Además de la conductividad, otro parámetro que caracteriza la calidad del agua es la turbidez. En 2016, Azman et al. [16] desarrollaron un sensor nefelométrico de turbidez para la monitorización continua de la calidad del agua. Los resultados obtenidos pueden ser visualizados directamente por el consumidor. Compararon el sensor desarrollado con el sensor de turbidez comercial (Hach 2100P) de la Unidad de Turbidez Nefelométrica (NTU). Este sistema se basa en el uso de LEDs que transmiten luz y receptores basados en resistencias dependientes de la luz. Se comunicaban con el procesador central (PIC 16F777) y un módulo (RS232). Arifin et al. en 2017 [17] desarrollaron un sensor de turbidez de diferentes longitudes de onda de fibra óptica polimérica para medir la turbidez. Utilizaron LED infrarrojo, fotodetector y fibra óptica polimérica. Como resultado, encontraron que el voltaje de salida se ve afectado por la longitud del sensor, si tiene o no un recubrimiento, la curvatura del sensor y la concentración de turbidez que presenta el agua.

En 2018, Wang et al. [18] desarrollaron un sensor de bajo costo para el monitoreo de la turbidez. El sensor se basa en la detección de luz transmitida y dispersa mediante el uso de LED infrarrojos de 850 nm (insertar nombre completo) y fotodetectores ortogonales duales. El diseño propuesto puede medir la turbidez dentro del rango de 0 a 1000 NTU. Mulyana y Hakim [19], en el mismo año, desarrollaron Arduino Leonardo, un sistema de monitoreo de la turbidez del agua. Utilizaron un fotodiodo y un LED infrarrojo para implementar el sensor de turbidez. Procesaron la señal analógica recibida



por el Arduino Leonardo. Descubrieron que el rango de detección de turbidez para el sensor desarrollado estaba entre 0 NTU y 40 NTU. Finalmente, Parra et al. [20] desarrollaron un sistema de turbidez para diferenciar el tipo de turbidez que era. El sensor implementado consta de cuatro LED. Además, está compuesto por fotodiodos y resistencias, que se encuentran a 180° . Utilizaron diferentes tipos de muestras; Isochrysis galbana, Tetraselmis chuii y sedimento. Finalmente, crearon un algoritmo para establecer la turbidez, la concentración de la fuente de turbidez y el origen de la misma. Como resultado, encontraron que se puede diferenciar el tipo de turbidez presente en el agua.

Mansor et al. en 2022 [21] desarrollaron un sistema de alerta para la contaminación del agua en los ríos. Con la implementación de este sistema, pretenden notificar a las autoridades cuando haya alguna anomalía. Los datos obtenidos a través de nitrógeno amoniacal, sólidos en suspensión y sensores bioquímicos de demanda de oxígeno se envían vía WiFi a la plataforma ThingSpeak. Encontraron que los errores del sensor de conductividad fueron del 6,84% y del 6,35%, comparándolo con el sensor de referencia. Finalmente obtuvieron lecturas precisas para el sensor de turbidez.

A diferencia de los trabajos presentados hasta el momento, se muestran estudios de calidad del agua y teledetección. En 2017, Sharaf El Din et al. [22] basaron su estudio en la determinación del deterioro de las aguas superficiales. Para ello, mapearon diferentes parámetros de calidad del agua superficial (SWQP, por sus siglas en inglés). Además, desarrollaron un método de modelado de inteligencia artificial para mapear este tipo de agua basado en la red neuronal de retro propagación para cuantificar las concentraciones de diferentes parámetros utilizando las imágenes del satélite Landsat 8. Llegaron a la conclusión de que el uso de esa red neuronal basada en Landsat 8 puede establecer concentraciones de diferentes SWQP a partir de imágenes de Landsat 8. Abdelmalik, en 2018 [23], quiso mostrar un mapa de distribución espacial en el lago Qaroun para cada parámetro de calidad del agua. Para ello, utilizó datos de la Radiación Espacial Avanzada de Emisión Térmica y Reflexión (ASTER). Entre los parámetros analizados en este estudio se encuentran la conductividad eléctrica del agua y la turbidez, así como la temperatura. Utilizó diferentes muestras para obtener la relación entre los parámetros de calidad del agua y los valores de ASTER. Como resultado, encontró una correlación significativa entre los valores observados y los datos de teledetección, con $R^2 > 0,94$ sig. $< 0,01$. Finalmente, en 2020, Sagan et al. [24] evaluaron la capacidad de la teledetección para evaluar la calidad del agua. Este estudio analizó más de 200 conjuntos de datos, incluyendo oxígeno disuelto, materia orgánica y conductividad eléctrica. Combinaron los datos tomados en el laboratorio y recogidos por sensores proximales distribuidos en diferentes zonas con imágenes hiperespectrales y datos satelitales. Como resultado, descubrieron que los parámetros ópticamente activos pueden ser obtenidos por los satélites. Sin embargo, esos parámetros ópticamente no activos se pueden estimar indirectamente, pero todavía hay problemas. Para ello, hay que utilizar el aprendizaje profundo.

Aunque todos estos estudios presentan avances significativos, también ofrecen algunas dificultades. Por ejemplo, la teledetección a menudo solo permite el monitoreo continuo del área de estudio. Además, las técnicas de procesamiento de imágenes suelen ser costosas y complicadas de manejar. Sin embargo, el presente estudio muestra que la combinación de diferentes sensores para el desarrollo de una sonda paramétrica de bajo coste puede permitir la evaluación de diferentes parámetros de calidad del agua. Con el nodo evolucionado de bajo costo, se permite el monitoreo continuo de la conductividad, así como la determinación de la turbidez. En este caso, la conductividad se mide como



salinidad, que también se conoce como sólidos disueltos totales (TDS). La salinidad o TDS se mide en g/L y es la suma de sólidos diluidos en el agua. En cuanto a la turbidez, se mide en términos de SST, que representa la cantidad de sólidos en el agua que no están diluidos ni sedimentados. El SST generalmente se mide en mg/L. La solución propuesta también permite establecer una cobertura más excelente debido a la gran cantidad de sensores desplegados en toda el área de estudio. Además, estas mediciones de temperatura se pueden compensar mediante el uso de un sensor de temperatura.

2.2 Comunicaciones Lora

LoRa (Long Range) está tomando relevancia debido a sus ventajas, particularmente en el envío de datos a través de largas distancias con un consumo de energía muy bajo. Cada tecnología tiene sus propias fortalezas y debilidades en términos de consumo de energía, eficiencia, alcance y capacidad. La Tabla I muestra las principales características de ambos protocolos en los términos antes mencionados.

LoRa es una modulación de espectro ensanchado patentada que pertenece a Semtech [25]. Mantiene las mismas características de baja potencia de la modulación FSK mientras aumenta el rango de comunicación. Según la banda de frecuencia (868 MHz en Europa), modula sus símbolos con un ancho de banda de 125 kHz, 250 kHz o 500 kHz (caso europeo) con diferentes factores de ensanchamiento (SF). Un SF más bajo permite aumentar la distancia máxima entre el transmisor y el receptor al disminuir la velocidad de datos.

Long Range (LoRa) es una tecnología inalámbrica en la que un transmisor de baja potencia envía pequeños paquetes de datos (entre 0,3 kbps y 5,5 kbps) a un receptor, normalmente a larga distancia. Estas características hacen que LoRa entre en la categoría de redes LPWAN [26]. Aunque el rango de cobertura de LoRa es amplio, depende en gran medida del entorno y de los materiales de construcción que lo componen. El alcance para entornos rurales es de aproximadamente 20 km. Sin embargo, en entorno urbano se reduce a 5 km. No obstante, algunas pruebas confirman que los datos se recibieron desde una distancia de cientos de kilómetros con visión directa adecuada y sin obstáculos en el área de Fresnel entre los dispositivos [27,28].

La tecnología LoRa Wireless fue desarrollada por la empresa francesa Cycleo. En 2012 la empresa estadounidense Semtech adquirió Cycleo, que ahora posee la patente de la parte de radio y la modulación LoRa y cuyos códigos son cerrados. Semtech proporciona licencias de propiedad intelectual a otras empresas, especialmente a fabricantes de componentes como HopeRF, Microchip, etc.

Por otro lado, los protocolos LoRaWAN están abiertos y definidos por LoRa Alliance, una organización sin fines de lucro fundada en 2015 con más de 500 empresas asociadas (IBM, Microchip, Orange, Cisco, etc.) creada con el compromiso de permitir y fomentar Implementaciones a gran escala de dispositivos LPWAN IoT que implementan sus estándares.

Iván Froiz-Míguez et al. presentó un sistema de seguimiento para que los trabajadores industriales evalúen su seguridad y salud en tiempo real [29]. Se implementaron sensores en dispositivos portátiles de IoT que se comunicaban con la puerta de enlace más cercana mediante tecnología LoRaWAN y LPWAN. La información recopilada se almacenó en ubicaciones distribuidas y se procesó utilizando blockchain para proporcionar inmutabilidad y trazabilidad a los datos al compartirlos con instalaciones médicas o compañías de seguros.



Boris Benites et al. propuso [30] un sistema de monitoreo de la escasez de agua destinado a lugares áridos y semiáridos. Se incorporaron sensores de humedad y anemómetros para enjambrear drones para monitorear zonas con altas concentraciones de agua. Estos drones reenviaron los datos mediante LoRaWAN al registro para su almacenamiento. Luego, los datos se transmitirían al nodo central para predecir la dirección del viento mediante técnicas de clasificación y regresión.

Emiliano Sisinni et al. diseñó [31] un extensor de alcance para IoT industrial LoRaWAN. Los autores sugirieron el uso de un enfoque de retransmisión de tramas para evitar comprometer las velocidades de datos más altas con una mayor sensibilidad. El prototipo fue creado a partir de hardware comercial. Se realizaron pruebas de cobertura midiendo el RSSI y el SNR para asegurar el correcto desempeño de la propuesta y demostrar su efectividad en el aumento del rango. Además, el extensor de alcance pudo funcionar con redes heredadas de LoRaWAN.

Algunos autores como Jia [32] presentan un sistema para monitorear la calidad del agua y del aire en humedales. El sistema utiliza tecnología LoRa para enviar los datos recopilados a la estación base y un algoritmo de fusión de datos para reducir la cantidad de datos que se envía. De esta forma, pudo mejorar el rendimiento de la red y reducir el consumo de energía. Yan-Ting et al. [33] proporcionan un sistema de monitoreo de la calidad del agua de Internet de las cosas (IoT) en tiempo real basado en LoRaWAN para monitorear la temperatura, la turbiedad, la conductividad y el pH del agua en el lago Dong en la Universidad Nacional Dong Hwa. Das et al. [34] implementan un sistema de verificación de la calidad del agua (pH, conductividad y temperatura) en tiempo real a través de varios sensores. A través de un sistema de alerta se envían mensajes a los funcionarios, en caso de detectar contaminación.

Algunos autores presentan estudios desde el punto de vista de la implementación de sistemas de bajo costo. Simitha et al. [35] desarrollan un sistema de monitoreo basado en IoT y WSN. A través de este sistema recopilan datos sobre la calidad del agua en tiempo real, para preservar y gestionar mejor los recursos hídricos. Su objetivo es lograr un sistema de bajo costo y bajo consumo de energía, utilizando módulos LoRa y el protocolo de comunicación LoRaWAN, para transmitir los valores de los sensores a la plataforma ThinkSpeak y realizar análisis adicionales. Entre los parámetros de calidad del agua monitoreados se encuentran la temperatura, el pH y la turbidez, mientras que el oxígeno disuelto (OD) se calcula utilizando la ecuación de dependencia de OD-temperatura.

Otros autores utilizan vehículos de superficie no tripulados (USV) para crear un sistema de seguimiento de la calidad del agua. Wu et al. [36] proponen un sistema móvil de monitoreo de la calidad del agua, basado en tecnologías LoRa e IoT utilizadas por USV, para monitorear varios parámetros del agua en el lago Dardanelle, Arkansas. Integran un conjunto de sensores en una plataforma móvil que envía los datos a una plataforma LoRa. Diseñaron un sistema de largo alcance y bajo costo para enviar datos a través de LoRa a la nube de The Things Network (TTN).

Capítulo 3. Propuesta de nodo sensor multiparamétrico

En esta sección vamos a proponer todas las tecnologías, diseños, arquitecturas, códigos utilizados en la elaboración de la demo y de las pruebas realizadas del sistema.

3.1 Sensores

En esta subsección vamos a detallar que sensores vamos a integrar en el proyecto y su funcionamiento como ejemplos de código para emplearlos.

3.1.1 Sensor de conductividad

El principio de funcionamiento del sensor de conductividad de elaboración propia se basa en dos bobinas acopladas. La primera bobina que la llamaremos primaria de unas 80 vueltas es alimentada por un generador de funciones portátil. La cual induce un campo electromagnético a la bobina secundaria de unas 40 vueltas. Es en esta etapa de transformación es donde medimos la variación de la señal inducida respecto al medio por el que se le induce. Con lo cual podemos ver los cambios de conductividad en nuestro caso el medio marino. Posteriormente antes de medir los cambios con el adc del Arduino pasamos la señal por etapas de rectificado y filtrado como podemos ver en la Figura 1. Por último, la placa se alimenta con un generador de funciones de amplitud 9V generando una señal sinusoidal de unos 340khz.

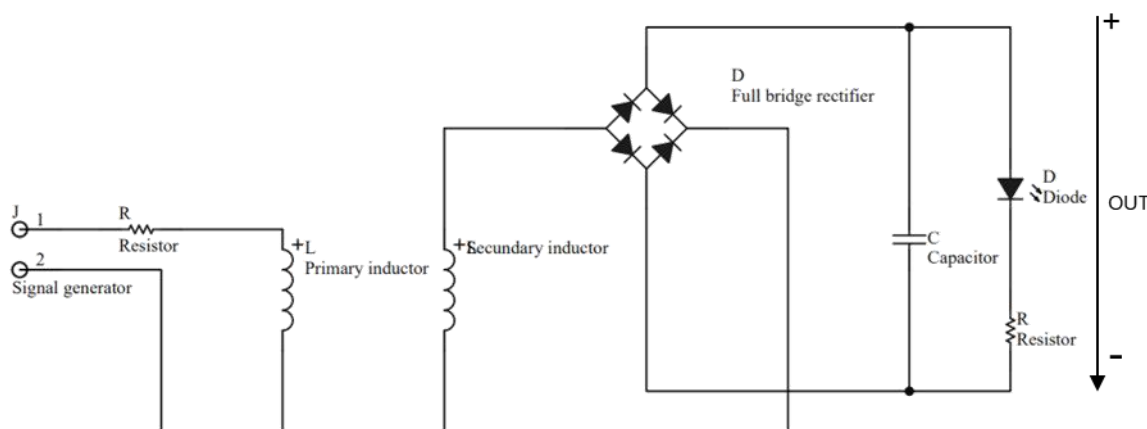


Figura 1 Diagrama electrónico del sensor de conductividad

Una vez implementado dicho esquema calibramos el sensor con distintas muestras de conductividad marina obteniendo un modelo de regresión lineal la cual describiremos en detalle como la hemos obtenido en el apartado de pruebas.

El circuito del sensor de conductividad irá integrado en la versión final en un circuito integrado para mejorar la señal recibida. Para ello se utilizarán las pistas rodeados y espaciados por pistas conectadas a tierra. Con ello conseguimos que las pistas se comporten como un cable coaxial y así mejorar la impedancia del circuito. En la Figura 2 podemos ver el esquema montado preparado para testear en una placa de desarrollo con el generador de funciones y las dos bobinas acopladas.

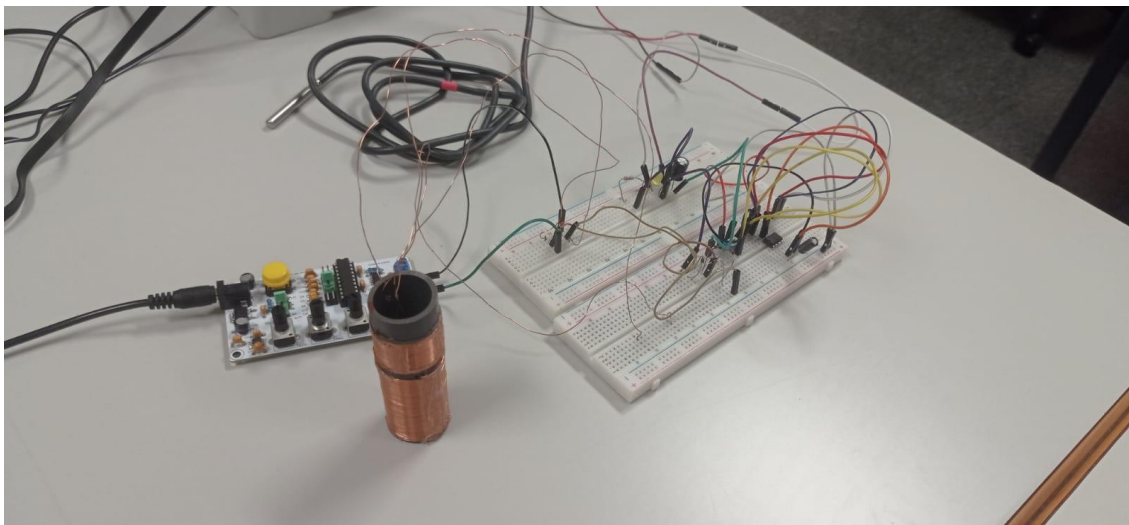


Figura 2 Sensor de conductividad montado en protoboard

Con dicho modelo de regresión lineal implementamos un código para la lectura por el adc y posterior conversión del valor en raw a S/cm que es la medida del sistema internacional a los medios conductivos. Quedando el código tal como en la Figura 3 donde se recogen el dato en bruto por el pin adc del ATmega2560 posteriormente se castea a un valor de voltaje en continua y también con el valor en bruto se obtiene el valor de conductividad en el sistema internacional.

```
90 float conductivity() {
91
92     float raw = adc1_get_raw((adc1_channel_t)6);
93     float V = raw / 4095 * Vcc;
94     Serial.print("Conductivity: ");
95     Serial.println(raw);
96     V = (100 * V) / 3.3;
97     Serial.println(V);
98     float conductividad = -256.9*raw + 579552;
99     return V;
100 }
```

Figura 3 Código para medir la conductividad en Arduino

3.1.2 Sensor de temperatura

El DS18B20 es un sensor de temperatura digital de alta precisión fabricado por Maxim Integrated [36]. Es un dispositivo muy popular debido a su precisión, bajo costo y facilidad de uso. Utiliza el protocolo OneWire para la comunicación con el microcontrolador, lo que permite conectar múltiples sensores a un solo pin del microcontrolador. En la Figura 4 podemos ver una ilustración de cómo es el sensor siendo de IP68.



Figura 4 Sonda de temperatura DS18B20

El DS18B20 utiliza un sensor de temperatura de tipo termómetro digital de 12 bits que convierte la temperatura en un valor digital que se puede leer fácilmente desde un microcontrolador. El sensor también cuenta con una memoria de 64 bits que almacena una identificación única, lo que permite la identificación única de cada sensor en un bus OneWire. El código para utilizarlo se puede ver en la Figura 5 donde se configura la sonda de la temperatura y se obtienen los valores de temperatura en grados Celsius y Fahrenheit en la parte del bucle principal.

```
1  #include <OneWire.h>
2  #include <DallasTemperature.h>
3
4  #define ONE_WIRE_BUS 2 // Pin digital al que está conectado el bus OneWire
5  OneWire oneWire(ONE_WIRE_BUS);
6  DallasTemperature sensors(&oneWire);
7
8  void setup() {
9      Serial.begin(9600);
10     sensors.begin();
11 }
12 void loop() {
13     sensors.requestTemperatures(); // Solicitar la lectura de temperaturas
14     float temperatureC = sensors.getTempCByIndex(0); // Obtener temperatura en grados Celsius
15     float temperatureF = sensors.toFahrenheit(temperatureC); // Convertir a grados Fahrenheit
16     Serial.print("Temperatura: ");
17     Serial.print(temperatureC);
18     Serial.print(" °C - ");
19 }
```

Figura 5 Ejemplo de código el sensor de temperatura DS18B20

3.1.3 Sensor de turbidez

El sensor virtual de turbidez utiliza 5 dimensiones que serán clasificadas por una red neuronal artificial. Estas dimensiones son el valor en bruto de la luz recibida por el fotodiodo. Siendo emitida por un diodo led a 180 grados de este. Teniendo como objetivo que la luz atraviese el medio.



Las dimensiones se componen de 5 componentes de luz; rojo, verde, azul, amarillo e infrarrojo. Una vez capturado el valor en bruto de los sensores por el microcontrolador se normalizan los datos para clasificarlos por una red neuronal artificial configurada para 5 entradas y 3 salidas sin contar las neuronas de la capa oculta.

Una vez clasificado el dato sabemos el origen de la turbidez que hay en el medio. A continuación, vamos a cuantificar la cantidad de turbidez que hay presente en el agua. Para eso hemos hecho una recta de regresión para cada tipo de turbidez que hay en el agua.

3.2 Microcontroladores y conexiones

3.2.1 Microcontrolador

El dispositivo utilizado es una placa microcontroladora basada en el ATmega2560. Tiene 54 pines de entrada/salidas digitales (de los cuales 15 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio.

Hemos elegido este microcontrolador por la robustez y variedad de pines. También porque es compatible con escudos de microcontroladores similares como el W5100 de Arduino UNO. Este lo usamos para mandar los datos por Ethernet y para guardar los datos en la SD. Otro factor es que es compatible con todas las librerías oficiales de Arduino y tiene mejores prestaciones en memoria de variables globales y del código en general.

3.2.2 Esquema de conexiones

El proyecto lleva incorporado un circuito impreso donde se agruparán las conexiones de los distintos sensores para posteriormente leerlos en el ATmega2560. Para ello se agruparán las conexiones a tierra y alimentaciones como los componentes de control de los sensores de dicha placa PCB. El esquema tiene que quedar como se muestra en la Figura 6.

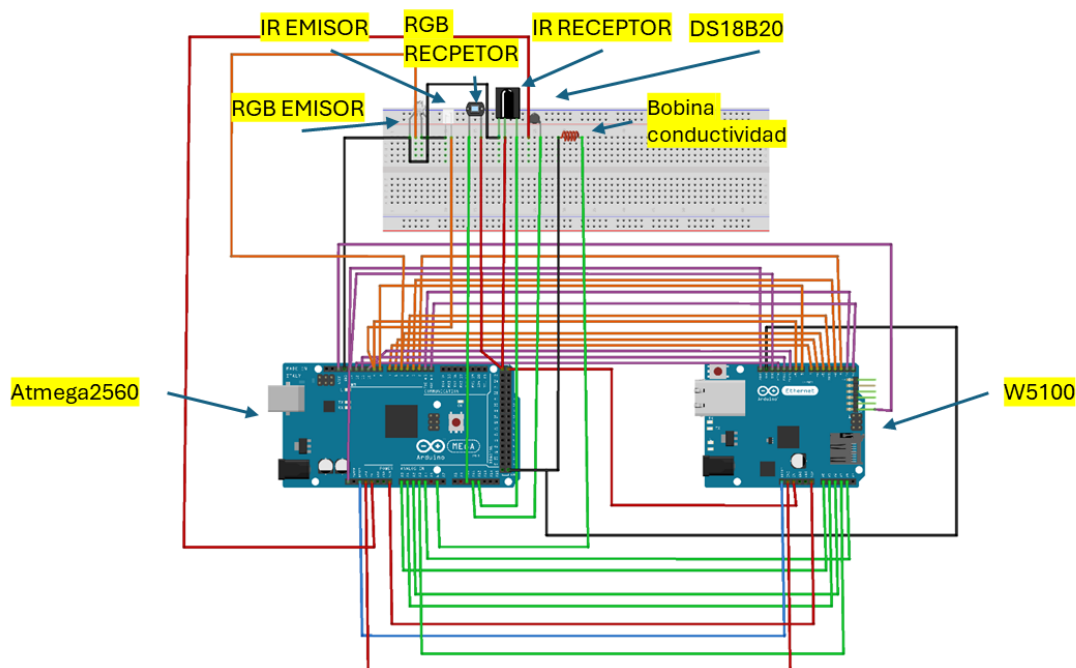


Figura 6 Esquema de conexiones empleado en el proyecto

También cabe destacar que el atmega2560 llevara acoplado el escudo W5100 para poder transmitir los datos por cable y guardarlos en la SD.

3.5 Encapsulado

Para el encapsulado se ha optado por una caja estanca IP65. Llevará un tubo sellado por dentro donde se le pasará una cadena para anclarlo al fondo oceánico y otro tubo transversal e inverso al del anclaje que será donde se tomen las medidas.

Toda la electrónica, baterías, etc. Está alojado en su interior y estará asilado con un gel dieléctrico llamado Magic blue. Las dos tapas y los agujeros producidos para integrar los elementos a el encapsulado también irán sellados para mejorar la impermeabilidad del producto.

3.6 Transmisión por cable

Para la transmisión por cable utilizo el escudo W5100 compatible con Arduino atmega2560 que utiliza las librerías oficiales de Arduino llamada Ethernet.h. También utilizo la librería SD.h para guardar los datos generados. Una vez instanciado el cliente declaro la IP del servidor al que voy a mandar los datos y el puerto en cuestión que en nuestro caso están dentro de la misma red.

En el setup del código inicializo la tarjeta SD y la Mac del dispositivo para que el Router asigne en la red una IP al dispositivo para la correcta transmisión de los datos. Si todo ha ido correctamente finalizo el setup sino aviso al usuario y lo vuelvo a intentar.



Una vez inicializado todo guardo los datos en la SD. Posteriormente creo un cliente HTTP con la IP y puerto del servidor. Si se conecta mando los datos al servidor en formato JSON y cerramos la conexión. Esto se lleva a cabo durante cada hora del día. En la Figura 7 podemos ver el pseudocódigo de la transmisión de los datos por cable.

Iniciar configuración:

```
Inicializar la comunicación serial a 9600 baudios
Inicializar la tarjeta SD en el pin 4
Establecer la dirección MAC arbitraria para la conexión Ethernet
Definir la dirección IP del servidor y el puerto del servidor
```

Fin de configuración

Inicio del bucle principal:

```
Si se logra guardar datos ficticios en la tarjeta SD:
  Si se logra enviar los datos al servidor:
    // Los datos se han guardado y enviado correctamente
  Fin si
Fin si
Esperar 5 segundos antes de la próxima escritura
```

Fin del bucle principal

Función para guardar datos ficticios en la tarjeta SD:

```
Abrir el archivo "datos.txt" en modo escritura en la tarjeta SD
Si el archivo se abre correctamente:
  Generar datos ficticios
  Escribir los datos en el archivo
  Cerrar el archivo
  Retornar verdadero
Sino:
  Retornar falso
```

Fin si

Fin de la función

Función para generar datos ficticios:

```
Generar valores aleatorios de luminosidad, temperatura, humedad, r,
g y b
```

```
Retornar los datos en forma de cadena
```

Fin de la función

Función para enviar datos al servidor:

```
Abrir el archivo "datos.txt" en modo lectura
```

```
Si el archivo se abre correctamente:
```

```
  Leer los datos del archivo
```

```
  Cerrar el archivo
```

```
  Conectar al servidor mediante la dirección IP y el puerto
  especificado
```

```
  Si se logra la conexión:
```

```
    Enviar una solicitud POST al servidor con los datos en
  formato JSON
```

```
    Cerrar la conexión
```

```
    Retornar verdadero
```

```
Sino:
```



```
        Retornar falso
    Fin si
Sino:
    Retornar falso
Fin si
Fin de la función
```

Figura 7 Ejemplo de código de la transmisión de los datos por cable

3.7 Transmisión inalámbrica

En la elaboración de este proyecto hemos estudiado todas las tecnologías de transmisión inalámbrica disponibles para elegir las más beneficiosas dependiendo de las condiciones de la localización del nodo.

3.7.1 Wi-Fi (IEEE 802.11) para comunicación en tierra

El cliente Wi-Fi funciona de manera muy similar al cliente Ethernet en Arduino. En este caso se utiliza la librería Wifi.h donde del mismo modo se instancia el SSID de la red Wi-Fi y la contraseña y se instancia un cliente nuevo. Y creando un cliente HTTPClient nos conectamos a la IP del servidor y el puerto en cuestión y si se conecta transmitimos los datos. Tal como se muestra la Figura 8.

```
Si WiFi está conectado entonces
    Crear un cliente WiFiClient
    Crear un cliente HTTPClient

    Inicializar una conexión HTTP hacia el servidor especificado
    ("http://192.168.0.100:3001/api/sensor/value/")

    // Especificar el tipo de contenido
    Agregar encabezado "Content-Type" con valor "application/json"

    // Formatear los datos a enviar en formato JSON
    Convertir los valores de las variables lumS, tempS y humS a cadenas
    de caracteres
    Concatenar los valores formateados en un string de datos JSON
    (dataString1) con la estructura adecuada

    // Realizar la solicitud POST al servidor con los datos en formato
    JSON
    Realizar una solicitud POST al servidor con los datos formateados
    en JSON y almacenados en dataString1

    Si la solicitud POST se realizó con éxito entonces
        Imprimir el código de respuesta HTTP obtenido
    Sino
        Imprimir un mensaje de error indicando el código de error
    obtenido
    Fin si

    // Liberar recursos utilizados por el cliente HTTP
    Finalizar la conexión HTTP
```



```
Sino
  Imprimir "WiFi Desconectado"
Fin si
```

Figura 8 Ejemplo de código para transmitir los datos por WIFI

3.7.2 LORA

La comunicación entre los nodos se establece mediante Lora con una potencia de antena de 20dbm y una frecuencia de canal de 433 MHz con un alcance teórico de 3km. Lora es una tecnología de comunicación inalámbrica de largo alcance utilizada para redes de sensores. Utiliza un sistema de reenvío de paquetes perdidos y se puede configurar para transmitir en un canal o dirección específica. Aunque es posible enviar paquetes en modo difusión. Además, para llevar la señal al módulo utilizamos paquetes UART al puerto serie de las placas, pudiendo modificar la velocidad en baudios, que en nuestro caso hemos fijado en 115200 bps como se puede ver en la Tabla 3.

Tabla 2 Configuración de los módulos Lora

Module configuration		
Feature	Bits	Configuration
AddH	1	
AddL	0	
Chan	25	435 MHz
SpeedParityBit	0	8N1
SpeedUARTDatte	111	115200bps
SpeedAirDataRate	111	19,2kbps
OptionTrans	0	Transparent transmission
OptionPullup	1	TXD, RXD, AUX are push-pulls/pull-ups
OptionWakeup	0	250ms
OptionFEC	0	
OptionPower	0	20dBm

Para la transmisión de los vídeos en Lora utilizamos dos nodos, uno actuando como emisor y otro como receptor. Ambos hardware están montados en dos cajas estancas IP55 alimentadas por una fuente de alimentación de 9V a 1A. La fuente está conectada a un convertidor DC-DC que alimenta un Atmega2560 en el caso del transmisor y un ESP32 en el caso del receptor. Estos nodos fueron seleccionados debido a su alta capacidad de procesamiento y para garantizar que la solución propuesta pueda usarse en redes heterogéneas en las que diferentes nodos tienen diferentes roles. Se seleccionó el Atmega2560 como emisor ya que tiene más pines, en este caso, 54 pines digitales y 16 pines analógicos. El módulo Lora utilizado es un Lora UART SX1278 con un alcance de 3 km. Los pines RX y TX están conectados al microcontrolador en modo pull-up. La conexión de los microcontroladores a los módulos Lora se realiza mediante una transmisión UART al segundo puerto serie de las placas. Finalmente, para transmitir el vídeo, hemos montado un escudo W5100 en el Atmega2560, donde podremos adquirir el vídeo desde la SD incorporada para posteriormente mandarlo por el módulo Lora al nodo

receptor tal como vemos en la Figura 9 donde podemos ver el interior de un nodo transmisor.

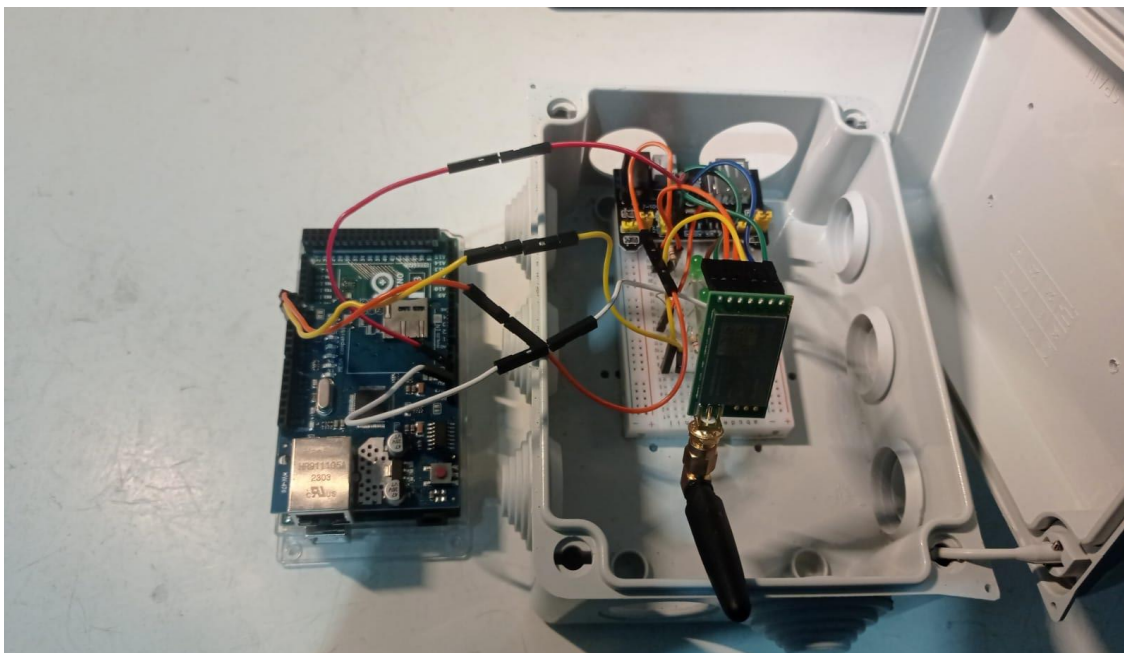


Figura 9 Nodo transmisor Lora montado

Como la carga útil de Lora es de 256 bytes, optamos por transmitir el vídeo en bloques de diferentes tamaños que la carga útil, aunque el tamaño del búfer se puede ajustar según los requisitos hasta un máximo de 4095. Intentar enviarlo byte a byte retrasa significativamente la transmisión. La configuración del módulo establece un canal de 435 MHz con una velocidad de bits de paridad 8N1. La velocidad de UART está establecida en 115200 bps, con una velocidad de datos aéreos de 19,2 kbps y una potencia de antena de 20 dBm.

Para la transmisión de video, el algoritmo propuesto se puede ver en la Figura 10. Se configuran temporizadores y banderas para determinar el inicio y el final de la transmisión, lo que permite calcular el tiempo de acceso al medio (MAT).

Inicio del Programa:

Definir constantes y variables globales necesarias para el programa.

Definir las funciones de impresión de parámetros y configuración del módulo Lora.

En la función setup():

- Inicializar la comunicación serial con una velocidad de 115200 baudios.
- Inicializar la comunicación serial 2 (Serial2) con una velocidad de 115200 baudios.
- Esperar la entrada del usuario para introducir la distancia de emisión.

- Inicializar el módulo Lora.
- Obtener la configuración actual del módulo Lora.
- Modificar los parámetros de configuración del módulo Lora según sea necesario.
- Establecer la nueva configuración del módulo Lora.
- Iniciar la tarjeta SD y comprobar si se inicializó correctamente.

En la función loop():

- Si SEND_VIDEO es verdadero:
 - Abrir el archivo de video desde la tarjeta SD.
 - Registrar el tiempo de inicio de la transmisión.
 - Leer y enviar bloques de datos del archivo de video a través de Lora.
 - Calcular y registrar el tiempo de fin de la transmisión.
 - Calcular y registrar el tiempo de procesamiento.
 - Calcular y registrar el tiempo de propagación basado en la distancia.
 - Imprimir los resultados de los tiempos de transmisión, procesamiento y propagación.
 - Desactivar el envío de video para evitar reenvíos continuos.

Figura 10 Pseudocódigo de la transmisión por Lora del emisor.

El nodo receptor comparte la misma configuración de módulo que el remitente para maximizar la velocidad del canal. El software del remitente incluye un oyente en el bucle para que, al llegar el paquete, recupere y muestre el contenido en la pantalla, permitiendo observar la longitud total del paquete correspondiente al original almacenado en la SD. El algoritmo para la recepción de vídeo se puede ver en la Figura 11.

Inicio del Programa:

Configurar los pines RX y TX para la comunicación con el módulo Lora a través de Serial2.
Inicializar el objeto LoRa_E32 y el objeto LiquidCrystal_I2C.
Declarar variables para registrar los tiempos de inicio y fin de varios procesos.
Definir las funciones de configuración e impresión de parámetros del módulo Lora.

En la función setup():

- Iniciar la comunicación serial con una velocidad de 115200 baudios.
- Iniciar la comunicación serial 2 (Serial2) con una velocidad de 115200 baudios, 8 bits de datos, sin paridad y 1 bit de parada.
- Inicializar el LCD.
- Encender el backlight del LCD.
- Inicializar el módulo Lora.
- Obtener la configuración actual del módulo.
- Imprimir la configuración actual del módulo.
- Modificar los parámetros de configuración según sea necesario.
- Establecer la nueva configuración del módulo Lora.
- Imprimir la configuración modificada del módulo Lora.
- Imprimir un mensaje de inicio del programa.

```
En la funció loop():  
- Verificar si hay un mensaje disponible para recibir en el módulo Lora.  
- Si hay un mensaje disponible:  
  - Registrar el tiempo de acceso al medio del receptor.  
  - Recibir el mensaje del módulo Lora.  
  - Imprimir el mensaje recibido.  
  - Registrar el tiempo de procesamiento de la recepción.
```

Figura 11 Pseudocódigo de la transmisión por Lora del receptor.

3.8 Servidor y base de datos

3.8.1 Servidor montado en Node Express

Para el servidor la estructura del código está montada en un modelo vista-controlador en el framework de express js. La idea del servidor es que un router Tp-Link tenga asignada una IP pública en un dominio web DNS como puede ser no-ip. Con eso aseguramos que cualquier persona que este alejada de la red pueda acceder al cliente mediante el navegador de su teléfono o su ordenador. Para brindar más seguridad el router lleva incorporado una PAT estática para que solo deje acceder peticiones HTTP por el puerto 80. La PAT transformará estas peticiones a la parte privada de la red al puerto 3000 que es la que por defecto usa el cliente REACT y únicamente comunique las peticiones con la IP del servidor.

Vamos a ver en detalle cómo funciona la programación del servidor que consta de 4 archivos:

3.8.1.1 server.js

En este archivo se declara y activa el servidor con la IP y el puerto que le hemos declarado. También se llama al archivo sensor route para que se activen las URL de las peticiones HTTP y que tipo de peticiones que son para facilitar la comunicación REST en la Figura 12 podemos ver el código.

```
1  const express = require('express');
2  const app = express();
3
4  const routes = require('./routes');
5  const bodyParser = require('body-parser');
6
7  var cors = require('cors')
8
9  app.use(cors())
10
11 app.use(bodyParser.json())
12
13
14 /// routes
15 app.use('/api', routes)
16
17 const port = process.env.PORT || 3001;
18 const host = '192.168.43.129';
19 app.listen(port, host, () => {
20   console.log(`Servidor corriendo en http://${host}:${port}`);
21 });
22
```

Figura 12 Código del archivo server.js

3.8.1.2 sensorRoute.js

Aquí podemos ver que tenemos dos peticiones registradas donde una es una GET para obtener todos los datos para mostrarlos en la web y otra POST para añadir datos. En la Figura 13 podemos ver el código.

```
1  const express = require('express');
2  const router = express.Router();
3  const sensorController = require('../controllers/sensorController');
4
5  router.post('/value', sensorController.uploadValue)
6  router.route('/').get(sensorController.getValues)
7
8  module.exports = router;
```

Figura 13 Código del archivo sensorRoute.js

3.8.1.3 sensorController.js

Cuando llega una solicitud la recoge y llama a la función de la lógica de negocio correspondiente. En esta llama a las funciones de la lógica de añadir un dato y a la de obtener todos los valores. En la Figura 14 podemos ver el código.

```
1  const httpStatus = require('http-status');
2  const { sensorService } = require('../services')
3
4  const sensorController = {
5    async uploadValue(req, res, next) {
6      try {
7        const value = await sensorService.addValue(req.body);
8        res.json(value);
9      } catch (error) {
10       next(error)
11     }
12   },
13   async getValues(req, res, next) {
14     try {
15       const values = await sensorService.allValues(req)
16       res.json(values)
17     } catch (error) {
18       next(error)
19     }
20   },
21 }
22
23 }
24
25 module.exports = sensorController;
```

Figura 14 Código del archivo sensorController.js

3.8.1.4 *sensorService.js*

Esta es la parte de la lógica de negocio donde se crea una conexión a la base de datos y se envían consultas SQL a la base de datos. Luego si es necesario se tratan los datos y se devuelven al cliente cuando se llama a alguna función de la lógica. En la Figura 15 podemos ver el código.

```
12 const addValue = async (body) => {
13   try {
14
15     var d = new Date();
16     var month = d.getMonth() + 1;
17     const fecha = d.getDate().toString() + "/" + month.toString() + "/" +
18     d.getFullYear().toString();
19     const hora = d.getHours().toString() + ":" + d.getMinutes().toString() +
20     ":" + d.getSeconds().toString();
21
22     connection.query(`INSERT INTO sensores (fecha, luminosidad,
23     temperatura, humedad, hora, r, g, b)
24     VALUES("${fecha}", "${Number(body[0].luminosidad)}",
25     "${body[0].temperatura}", "${body[0].humedad}", "${hora}", "${body[0].r}",
26     "${body[0].g}", "${body[0].b}")`,
27     (err, rows, fields) => {
28       if (err) throw err
29     })
30   }
31 } catch (error) {
32   throw error
33 }
34 }
35 }
36
37 const allValues = async (req) => {
38
39   try {
40     const values = await query("SELECT * FROM sensores")
41     return values;
42   } catch (error) {
43     throw error
44   }
45 }
```

Figura 15 Código del archivo sensorService.js

3.8.2 Base de datos

La base de datos está montada en MySQL con un motor MariaDB administrada con phpmyadmin. Consta de una tabla llamada prueba donde envió un dato y una id para probar que el servidor funcionara en la versión mínima posible. Y luego en la Figura 16 podemos ver el diseño de la base de datos donde en la tabla sensores que es la que funciona en la versión final con la lógica con estos nueve campos:

- Id: se genera automáticamente cada vez que se inserta un nuevo dato. Es autoincrementale.
- Fecha: fecha de cuando se ha obtenido la medición por el servidor.
- Conductividad: valor de conductividad del nodo en la medición.
- Temperatura: valor de temperatura del nodo en la medición.
- Turbidez: valor de turbidez infrarroja del nodo en la medición.
- Red: valor de la componente roja de turbidez del nodo en la medición.
- Green: valor de la componente verde de turbidez del nodo en la medición.
- Blue: valor de la componente azul de turbidez del nodo en la medición.
- Hora: hora en la que se ha hecho la medición.



Field	Type
id	int(250)
fecha	varchar(12)
luminosidad	float
temperatura	float
humedad	float
hora	varchar(8)
r	varchar(255)
g	varchar(255)
b	varchar(255)

Figura 16 Diseño de la base de datos

3.9 Cliente web

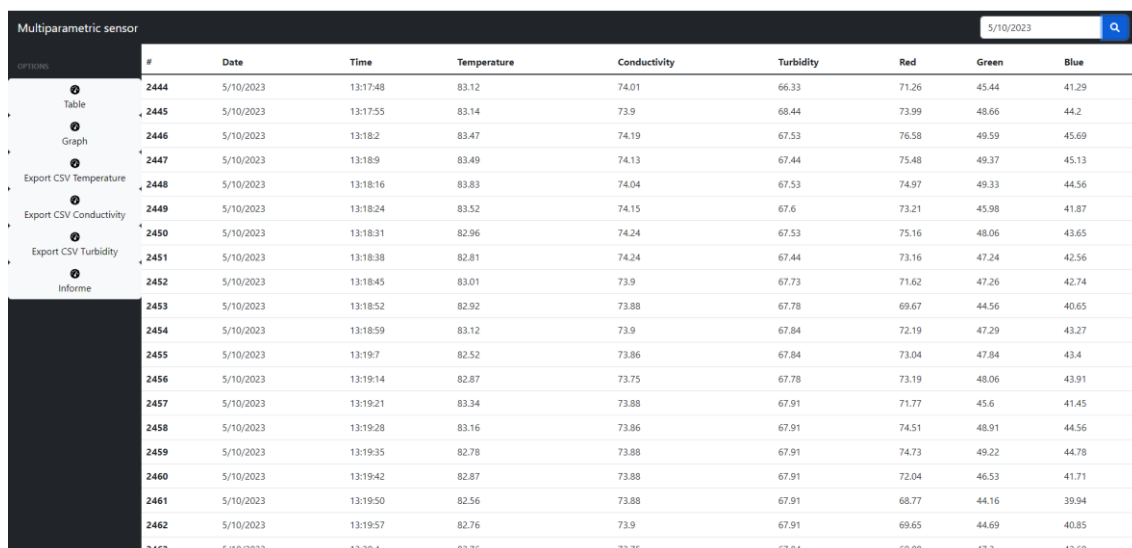
El cliente como ya hemos mencionado está montado en REACT con una estructura en las carpetas por tipo de fichero. En nuestro caso components, hoc, store, utils y el index.js y un archivo routes.js.

- Components: Aquí se desempeña todo el front-end de toda la web como son los elementos que se muestran y sus funcionalidades.
- Hoc: Aquí hay alojados elementos estáticos como puede ser el header
- Utils: Aquí están alojados elementos estáticos como puede ser las filas de la tabla, la tabla de los informes y las gráficas.
- Store: Aquí está implementada la lógica fake de la aplicación con accionadores y reductores para comunicarse el cliente con el servidor.
- Index.js: Aquí se declaran los elementos principales de la web.
- Routes.js: Aquí se declaran las rutas de navegación de la web.

Ahora en las siguientes subsecciones vamos a exponer las funcionalidades de la web.

3.9.1 Tabla principal

En esta ventana disponemos de un histórico de todos los datos recogidos en una tabla. Pudiendo filtralos por el día en el que se hizo la medición como se puede ver en la Figura 17.



#	Date	Time	Temperature	Conductivity	Turbidity	Red	Green	Blue
2444	5/10/2023	13:17:48	83.12	74.01	66.33	71.26	45.44	41.29
2445	5/10/2023	13:17:55	83.14	73.9	68.44	73.99	48.66	44.2
2446	5/10/2023	13:18:2	83.47	74.19	67.53	76.58	49.59	45.69
2447	5/10/2023	13:18:9	83.49	74.13	67.44	75.48	49.37	45.13
2448	5/10/2023	13:18:16	83.83	74.04	67.53	74.97	49.33	44.56
2449	5/10/2023	13:18:24	83.52	74.15	67.6	73.21	45.98	41.87
2450	5/10/2023	13:18:31	82.96	74.24	67.53	75.16	48.06	43.65
2451	5/10/2023	13:18:38	82.81	74.24	67.44	73.16	47.24	42.56
2452	5/10/2023	13:18:45	83.01	73.9	67.73	71.62	47.26	42.74
2453	5/10/2023	13:18:52	82.92	73.88	67.78	69.67	44.56	40.65
2454	5/10/2023	13:18:59	83.12	73.9	67.84	72.19	47.29	43.27
2455	5/10/2023	13:19:7	82.52	73.86	67.84	73.04	47.84	43.4
2456	5/10/2023	13:19:14	82.87	73.75	67.78	73.19	48.06	43.91
2457	5/10/2023	13:19:21	83.34	73.88	67.91	71.77	45.6	41.45
2458	5/10/2023	13:19:28	83.16	73.86	67.91	74.51	48.91	44.56
2459	5/10/2023	13:19:35	82.78	73.88	67.91	74.73	49.22	44.78
2460	5/10/2023	13:19:42	82.87	73.88	67.91	72.04	46.53	41.71
2461	5/10/2023	13:19:50	82.56	73.88	67.91	68.77	44.16	39.94
2462	5/10/2023	13:19:57	82.76	73.9	67.91	69.65	44.69	40.85
2463	5/10/2023	13:20:4	82.76	73.75	67.84	69.08	47.2	42.69

Figura 17 UX tabla principal de la web

3.9.2 Gráfica general

En esta ventana disponemos de un histórico de todos los datos recogidos en formato de gráfica. Pudiendo seleccionar las medidas que queramos visualizar como se puede ver en la Figura 18.

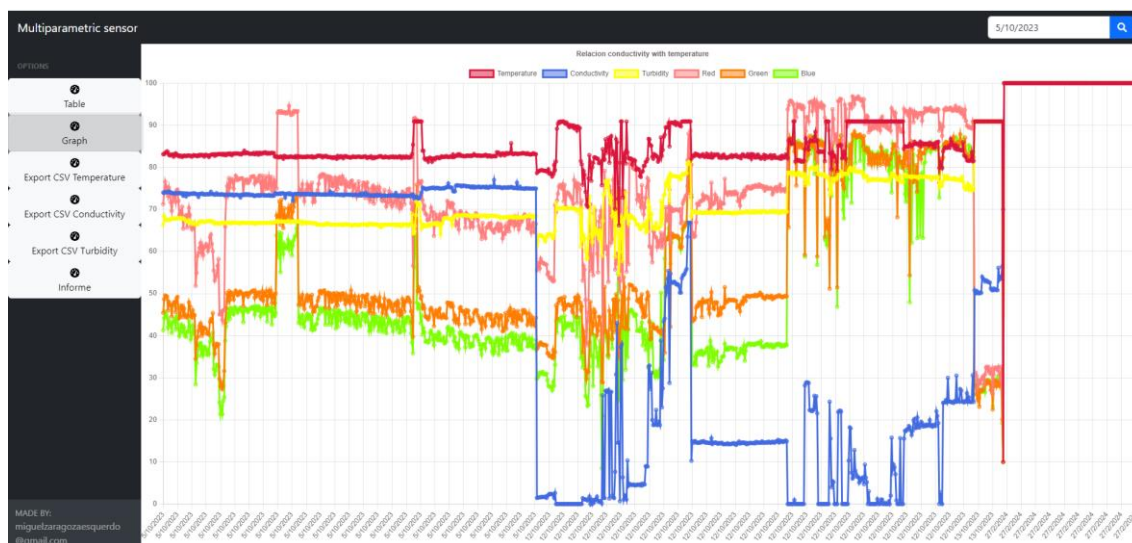


Figura 18 UX gráfica general de la web

3.9.3 Exportación de los datos a formato Excel

En la barra de opciones podemos pulsar las opciones de exportar algunas de las variables o campos registrados en formato csv para poder modificarlos en Excel como se puede ver en la Figura 19.

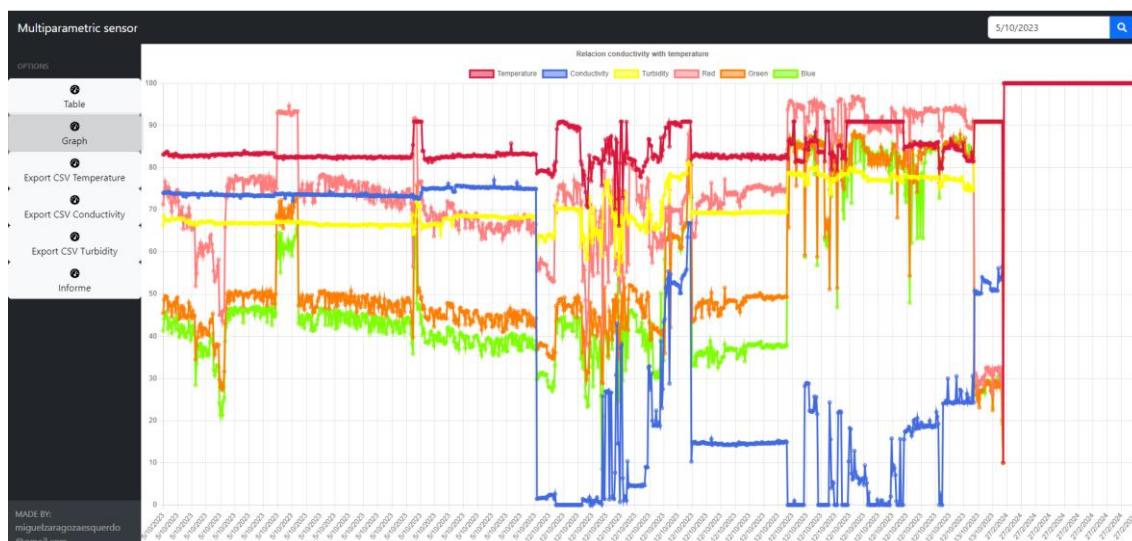


Figura 19 UX de la exportación de los datos en Excel de la web

3.9.4 Informes

Por último, está el apartado de informes donde podemos ver un procesamiento de datos registrados que nos muestra las condiciones en la que está la situación en la zona costera como se puede ver en la Figura 20.

Datos sobre el estado de la atmósfera

Fechas	Humedad	Temperatura	Masa de Aire	Estado atmosférico
Fechas	Humedad	Temperatura	Masa de Aire	Estado atmosférico

Figura 20 UX de la zona de informes de la web

Capítulo 4. Desarrollo

En esta sección vamos a tratar como se ha adaptado todos los elementos de la propuesta en la demo que se ha utilizado para hacer todas las pruebas elaboradas en el desarrollo de este trabajo de final de grado.

En la Figura 21 podemos ver como se disponen los elementos del observatorio marino en el nodo sensor con los diferentes sensores físicos. En ella resalta en la parte central un tubo que atraviesa el encapsulado en el cual cuando la sonda esta esté fondeada circulará el agua por el interior. Por tanto, en la parte superior están situados los sensores de conductividad y turbidez que a través del tubo transparente capturarán los datos.

A la parte derecha podemos ver la placa PCB que se encarga de filtrar electrónicamente los valores en continua para posteriormente mandarlos al Arduino. La cual irá alimentada por un generador de funciones portátil que a su vez está alimentada por una batería.

Por último, a la parte izquierda del tubo podemos ver el microcontrolador, el lector de tarjetas SD y una protoboard para las conexiones. Lo cual todo ello da cobertura electrónica al proyecto para su correcto funcionamiento. Y más a la derecha se encuentra el sensor de temperatura que irá directamente en contacto al agua y conectado a la demo.

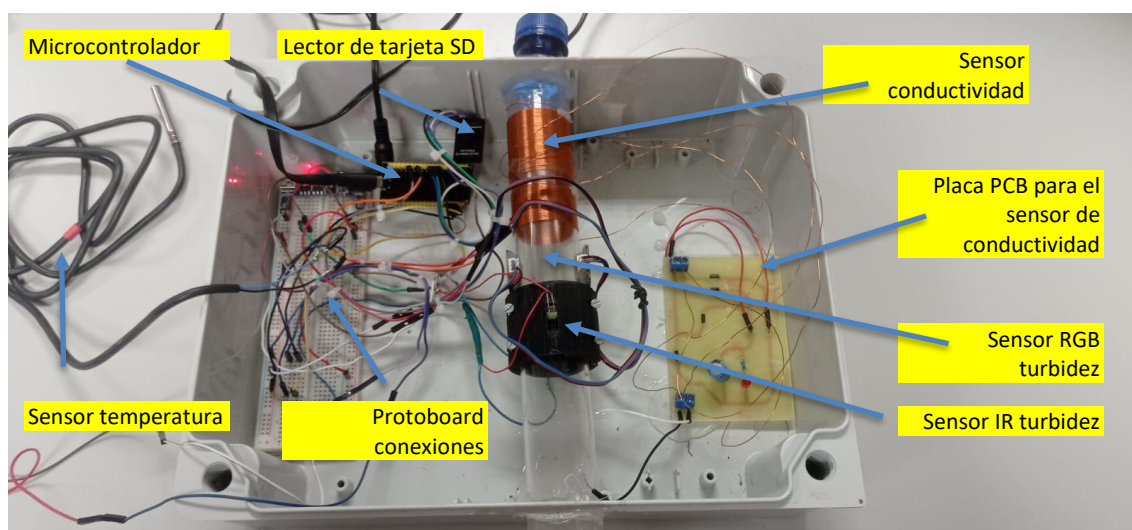


Figura 21 Disposición de los elementos del observatorio en la demo

4.1 Tubo para obtener las medidas del agua del mar

Un aspecto importante en el proyecto es como obtener las medidas del océano. Para ello se ha ideado que en el encapsulado que irá totalmente sellado atraviese por dentro un tubo donde circule el agua por el interior para obtener las medidas.

Como se puede ver en la Figura 22 el tubo es totalmente transparente ya que es necesario para que los sensores ópticos de turbidez funcionen correctamente. Ya que Los diferentes colores de luz deben atravesar el tubo a través de su diámetro para capturar las variables de turbidez en el agua marina.

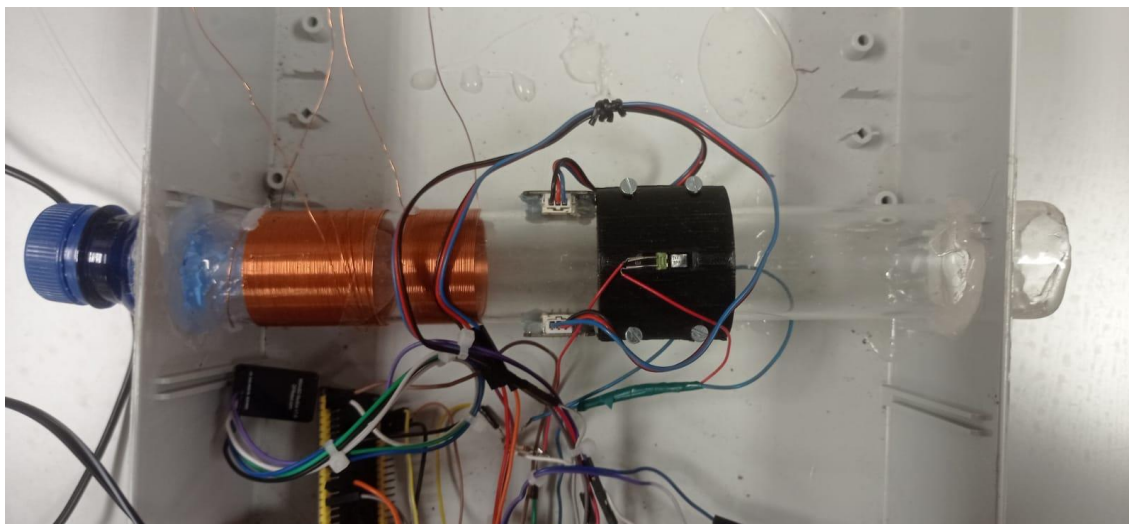


Figura 22 Prototipo del tubo donde se toman las medidas

4.2 PCB sensor de conductividad

Para dar robustez al proyecto se optó por la opción de crear circuitos impresos para los circuitos electrónicos. Como se puede ver en la Figura 23 la PCB hace de etapa de valores de filtrado, alimentación y conversión de los valores obtenidos en la bobina en el tupo para posteriormente mandarlos al convertidor analógico-digital del microcontrolador.

El circuito impreso es de elaboración propia y se hizo con la combinación de técnicas de grabado, abrasión del cobre con una solución de ácido clorhídrico y agua oxigenada que eliminaba el cobre de la placa que no estaba grabada. Posteriormente se hicieron los agujeros de las conexiones de los componentes y se soldaron todos los componentes.

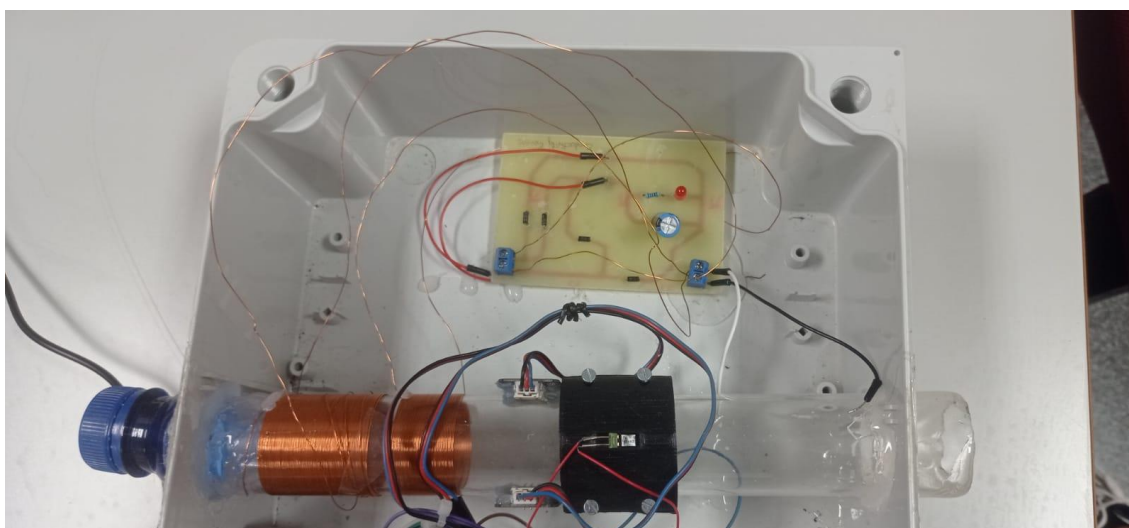


Figura 23 Circuito impreso del sensor de conductividad

4.3 Adaptación de la bobina de conductividad al tubo

La adaptación de la bobina de conductividad al tubo es un aspecto clave en el proyecto ya que al ser un sensor de elaboración propia se probaron muchas configuraciones. La

ideal es que la bobina estés por encima del tubo y que el campo electromagnético sea inducido a todo el medio que queremos leer. Como se puede ver en la Figura 24 se bobinó al tubo las bobinas inductoras e inducidas con alambre de cobre esmaltado de 0.5 mm de diámetro siendo esta la mejor configuración y dando buenos resultados en el laboratorio.

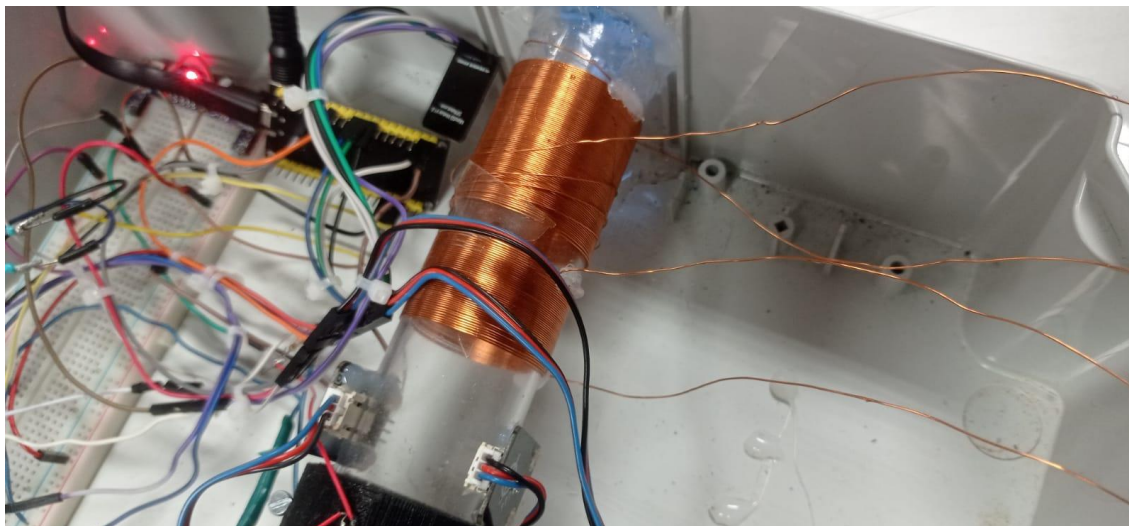


Figura 24 Bobina primaria y secundaria integrada al tubo

4.4 Abrazadera para el sensor de IR de turbidez

Para adaptar el sensor de infrarrojos de turbidez al tubo que atraviesa a la demo hemos diseñado una abrazadera hecha e imprimida en 3D con filamento PLA. Ella se dispone de dos piezas simétricas unidas con tornillería donde hay dos hendiduras para colocar el foto-emisor de infrarrojos y otra para el foto-receptor de infrarrojos consiguiendo adaptar los componentes electrónicos al tubo tal como se puede ver en la Figura 25.

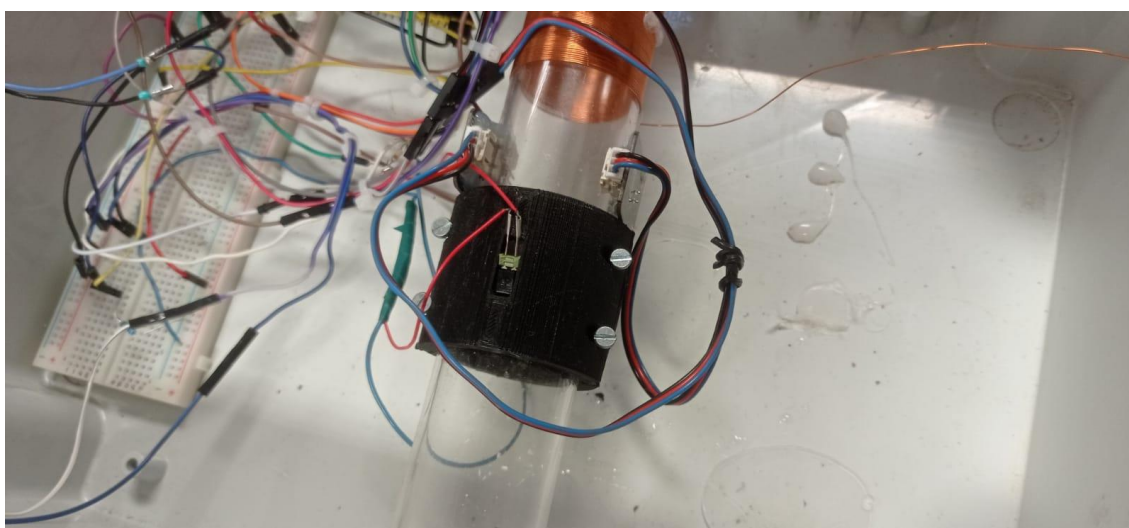


Figura 25 Abrazadera para el sensor de IR de turbidez

4.5 Adaptación del sensor RGB de turbidez

Tal como se puede ver en la Figura 26 para el sensor de turbidez también se integran sensores foto-emisores y foto-receptores en el tubo. Los cuales emiten la luz a través del tubo de un extremo de el a otro. Como son sensores con un encapsulado más robusto que los de infrarrojos no se ha requerido de utilizar una abrazadera. Y por tanto van pegados cuidadosamente con termofusible al tubo estando dispuesto uno y el otro a las partes opuestas de su diámetro.

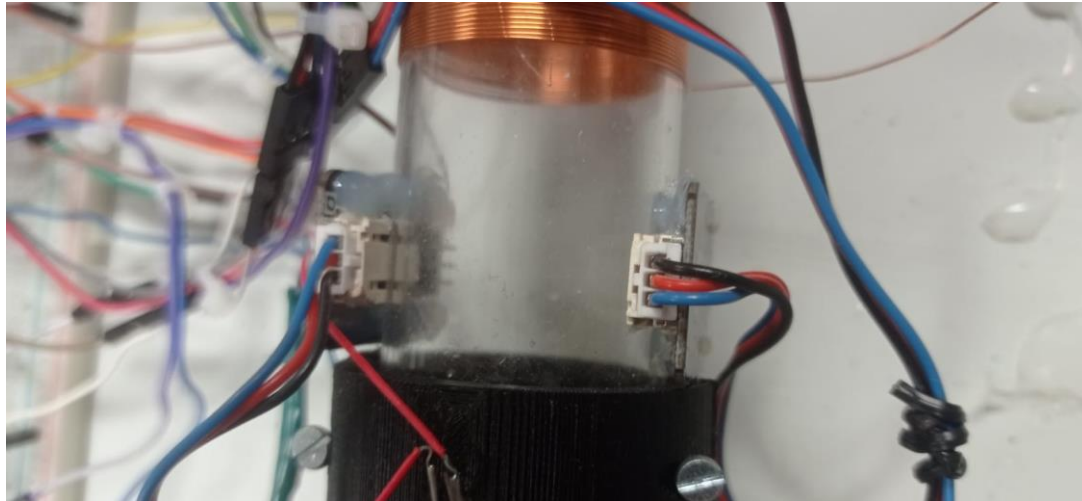


Figura 26 Adaptación del sensor RGB de turbidez

Capítulo 5. Pruebas

Este capítulo muestra las diferentes pruebas realizadas con los sensores, de manera individual, con una primera fase de calibrado y el uso de todos los sensores en conjunto.

5.1 Calibrado de sensores

En esta subsección se detallan las pruebas realizadas para la calibración de los sensores que forman parte del observatorio marino.

5.1.1 Calibrado del sensor de conductividad

En esta subsección, describimos el proceso para evaluar el desempeño de los sensores propuestos. En primer lugar, se describe el equipamiento de laboratorio utilizado. Luego, se describen las muestras generadas para calibrar y validar los sensores.

5.1.1.1 Equipo de laboratorio

Se utilizó equipo profesional para medir la conductividad del agua. Se ha seleccionado el Crison Basic 30 para medir la conductividad de las muestras. El equipo ha sido previamente calibrado con la solución estándar de 1413 $\mu\text{S}/\text{cm}$. La solución estándar y las muestras se mantuvieron a temperatura ambiente, 25 °C.

Se utilizó una balanza analítica para medir el agua y la sal de las muestras. El modelo utilizado fue el Labotronic Sartorius Entris Advanced. Se utilizó un cristal de reloj para pesar la sal. Mientras tanto, se pesó el agua en el vaso utilizado para contener las muestras.

5.1.1.2 Muestras generadas

Se ha llevado a cabo el siguiente proceso para evaluar el rendimiento del sensor propuesto y su circuito de acondicionamiento. En primer lugar, se generaron 5 muestras para el proceso de calibración. Luego, se crearon 2 muestras adicionales con valores cercanos a la salinidad del mar para verificar la curva de calibración. Las muestras se realizaron mezclando agua destilada y NaCl puro, la principal sal del mar.

Las muestras de calibración tienen valores de salinidad de 0 a 40 g/L y se pueden ver en la Tabla 5. Estos valores para la calibración verificarán si el sensor tiene un rendimiento estable en un amplio rango de salinidad. Además, cubre una amplia variedad de valores potenciales de baja salinidad debido a penachos de ríos o manantiales submarinos. Por otro lado, las muestras de verificación tienen valores entre 35 y 37 g/L, que son los valores habituales en la zona costera. La sal añadida en estas dos muestras fue de 17,5 y 18,5 g en los 500 ml de agua destilada.

Tabla 3 Muestras de calibración

ID	Agua (mL)	Sal (g)	Salinidad (g/L)
1	500	0	0
2	500	5	10
3	500	10	20
4	500	15	25
5	500	20	40

5.1.1.3 Métricas

El modelo de calibración se calcula mediante gráficos estadísticos. Se podrían calcular diferentes modelos según las tendencias observadas en los datos. Para evaluar el rendimiento del sensor y la calibración propuesta, utilizamos una serie de métricas. Las métricas utilizadas incluyen el coeficiente de correlación, el R2, el error absoluto promedio (MAE) y el error cuadrático relativo medio (MRE). El modelo de calibración se calcula utilizando el software Statgraphics.

5.1.1.4 Proceso de recopilación de datos

Los valores utilizados tanto para la calibración como para la verificación de los sensores se obtuvieron siguiendo el mismo procedimiento. Para cada muestra, el sensor se sumerge y, después de 30 segundos, se guardan los datos sin procesar. En cada muestra se recogen un total de 10 valores brutos. Estos valores se analizan y los valores promedio se calculan después de que se hayan eliminado los posibles valores atípicos. Los valores promediados para cada muestra son los datos incluidos para el cálculo del modelo de calibración y su validación.

5.1.1.5 Modelo de calibración

En primer lugar, se proporcionan los análisis de conductividad de las muestras para garantizar que la adición de sal se haya realizado correctamente. Los valores de conductividad se pueden ver en la Tabla III. Existe una correlación entre los valores de salinidad y conductividad de 0,9991, lo que confirma que las muestras se generaron correctamente.

A continuación, se presentan los datos recopilados. Todos los datos sin procesar se representan en la Tabla 6 como diagramas de caja. Es posible ver que las muestras 3 y 4 presentan valores atípicos que no están presentes en los análisis posteriores. Con esta visualización inicial de datos, se ve una tendencia clara entre las muestras y los valores brutos recopilados.

Tabla 4 Conductividad de muestras de calibración y validación

ID	Uso	Salinidad (g/L)	Conductividad (mS/cm)
1	Calibración	0	14.5
2	Calibración	10	26.6
3	Calibración	20	33
4	Calibración	25	38.5
5	Calibración	40	44.2
6	Calibración	35	48.9
7	Calibración	37	51.2

Después de eliminar los valores atípicos de las muestras 3 y 4, se calculan los modelos de regresión. Para seleccionar los modelos más adecuados se realiza la Comparación de Modelos Alternativos. Los resultados de esta comparación se pueden ver en la Tabla 7. Si bien la regresión lineal no es la que tiene mayor coeficiente de correlación, dada la

pequeña diferencia con el modelo superior, ambos modelos se comparan en los siguientes párrafos por su simplicidad.

Entre los dos modelos de calibración estudiados, en primer lugar, presentamos el gráfico X-Y para el modelo Inverso de Y, que se puede observar en la Figura 27. Cabe señalar que para la obtención de los modelos se utilizan valores promedio. En la Figura 28, se puede ver que, aunque el modelo es inverso al modelo Y, la apariencia es casi un modelo lineal. La ecuación de este modelo se puede ver en la Ecuación 1. La representación del modelo lineal se muestra en la Figura 29. Las predicciones y los intervalos de confianza se representan en líneas grises y verdes en este caso. Este modelo sigue la ecuación que se muestra en la Ecuación 2.

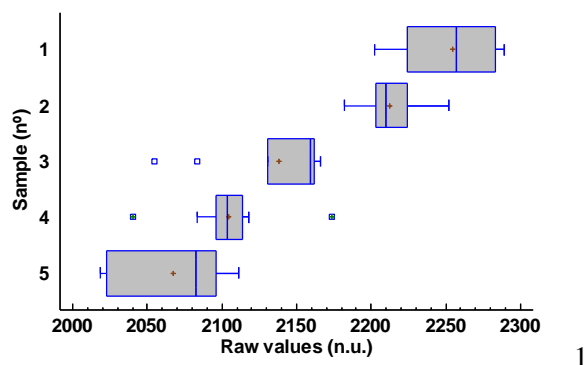


Figura 27 Análisis descriptivo para las muestras de la calibración

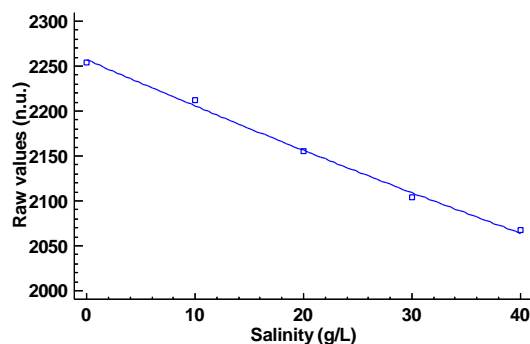


Figura 28 Modelo inversa de Y para salinidad

Tabla 5 Tabla de los diferentes de coeficientes de correlación por modelo

ID	Nombre del modelo	Coefficiente de correlación
1	Inverrsa de Y	0.9981
2	Exponencial	-0.9980
3	Raiz cuadrada de Y	-0.9979
4	Lineal	-0.9978
5	Raiz de Y	-0.9974
6	Raiz Y - X Raiz cuadrada	-0.9583
7	Inversa Y - X Cuadrado	0.9559

ID	Nombre del modelo	Coefficiente de correlación
8	Raiz cuadrada de X	-0.9555
9	Doble raiz cuadrada	-0.9541
10	Logaritmo Y – Raiz cuadrada de X	-0.9526
11	Logaritmo de Y - X Cuadrado	-0.9524
12	Raiz cuadrada de X - X Cuadrado	-0.9506
13	Inversa Y – Raiz cuadrada de X	0.9496
14	Cuadrada de X	-0.9488
15	Doble cuadrado	-0.9451

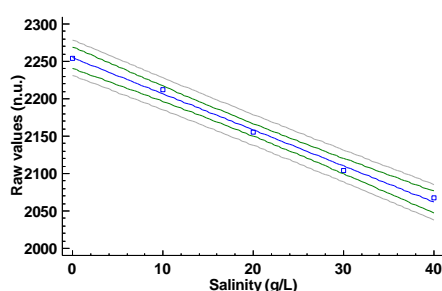


Figura 29 Modelo de regresión lineal para salinidad

$$Raw = \frac{1}{0.000443074 + 0.0000010348 \times Salinidad \left(\frac{g}{L}\right)} \quad (1)$$

$$Raw = 2254.8 - 4.82 \times Salinidad (g/L) \quad (2)$$

Para validar los modelos de regresión obtenidos, los dos valores verdaderos de salinidad de las muestras de validación se comparan con los valores predichos. Los valores verdaderos para las muestras son 35 y 37 g/L.

Los valores de salinidad pronosticados aplicando el modelo de calibración basado en la ecuación Inversa de Y son 34,43 y 37,09 g/L. Mientras tanto, cuando se utiliza el modelo lineal, los valores predichos son 34,39 y 36,89 g/L. Las métricas para evaluar el mejor modelo se pueden ver en la Tabla X. Estas métricas incluyen no solo las vinculadas a la validación sino también las de calibración, como R2.

Tabla 6 Resultados de la comparación de modelos alternativos

Metricas	Inversa de Y	Modelo lineal
R2	99.6	99.5
MAE	-0.24	-0.35
MRE	-0.69%	-1.01%

Las métricas presentadas en la Tabla 8 indican una alta similitud entre los modelos con un valor ligeramente mejor para el modelo Inverso de Y. Sin embargo, en este artículo se propone y compara una última métrica. Estas métricas reflejan el cambio mínimo de salinidad que el sistema puede diferenciar en función del carácter discontinuo de los datos de guerra y el modelo seleccionado. Centrándonos en la región del modelo cercano a la salinidad del mar de las muestras de validación, se seleccionan dos valores brutos: 2078 y 2077. Luego, aplicamos los modelos de regresión para ambos valores. Para el modelo Inverso de Y, las salinidades previstas serán 36,87 y 37,09, que corresponden a una diferencia de 0,22 g/L. Cuando se aplica el modelo Lineal los valores de salinidad son 36,89 y 36,68, una diferencia de 0,21 g/L. Esto significa que el modelo lineal es más sensible en la región de interés del modelo de regresión. Por tanto, este es el modelo preferido.

5.1.2 Calibrado del sensor de temperatura

Para el calibrado de temperatura se ha comparado la sonda DS18B20 con una sonda comercial de laboratorio. En la Figura 30 podemos ver que el comportamiento de las dos es muy similar respecto a las variaciones de temperatura medida en el laboratorio entre unos 44 grados hasta los 5 grados.

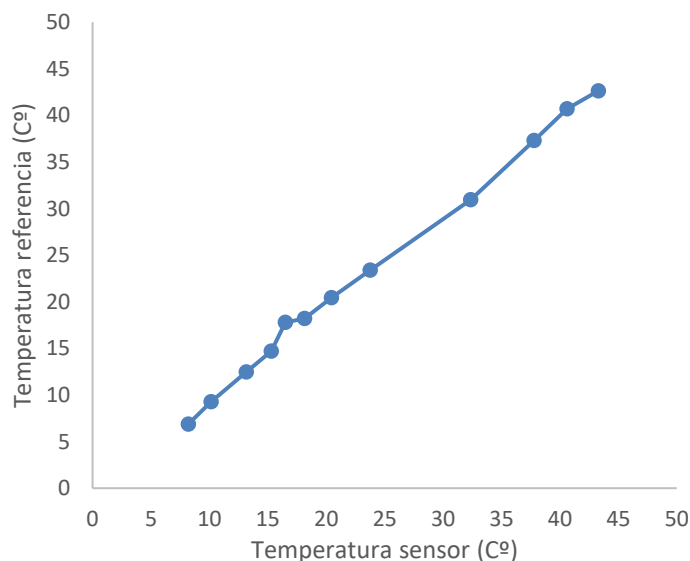


Figura 30 Comparación de la sonda de laboratorio respecto a la sonda DS18B20

5.1.3 Calibrado del sensor de turbidez

Para las muestras utilizadas tal como se ve en la Figura 31 se ha aplicado un análisis descriptivo. Podemos ver en el que las muestras tienen dos tendencias claras para emplear en el calibrado. Eso nos indica que hay dos orígenes distintos en las muestras utilizadas y por tanto debemos sesgar los datos obtenidos por la red neuronal en dos modelos de regresión lineal distintos una vez la hemos clasificado.

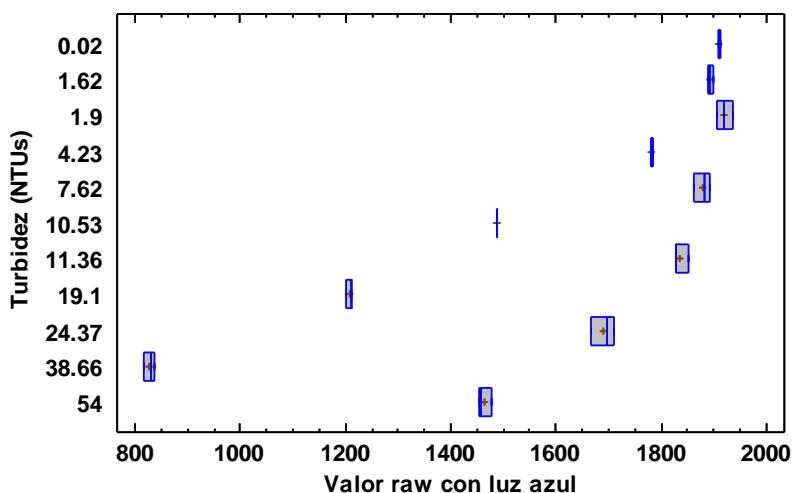


Figura 31 Análisis descriptivo de los datos

Como podemos ver en la Figura 32 tenemos dos clases. La primera es la materia vegetal en descomposición y la otra es materia vegetal fresca. Con los valores de medidos de luz rojo y verde en bruto ya podemos discernir la ubicación clase en distintos grupos.

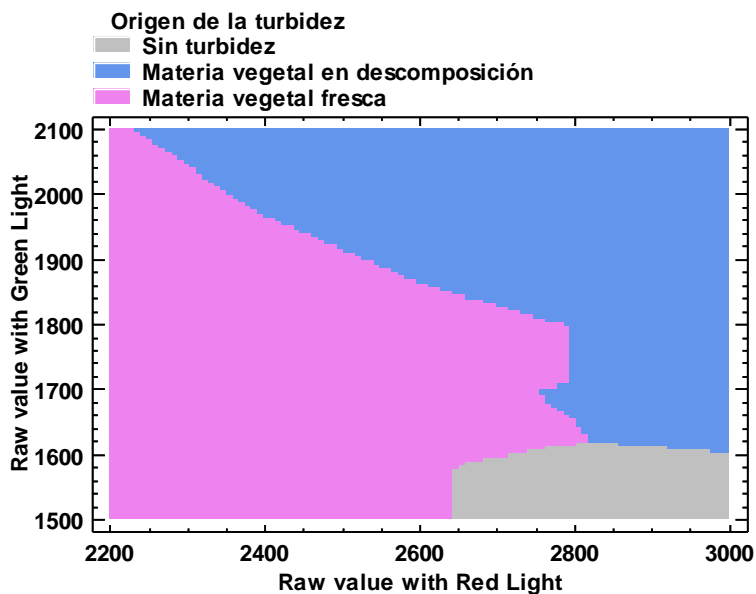


Figura 32 Representación de los grupos de turbidez

Una vez tenemos la separación de los datos en bruto por clase aplicamos la recta de regresión tal como se puede ver en la Figura 33. En este caso se trata de la materia vegetal en descomposición dado por la Ecuación 3 que posteriormente aplicaremos en el código para obtener el valor de NTUs desde los datos en bruto obtenidos por el sensor de turbidez en concreto en la banda de luz azul.

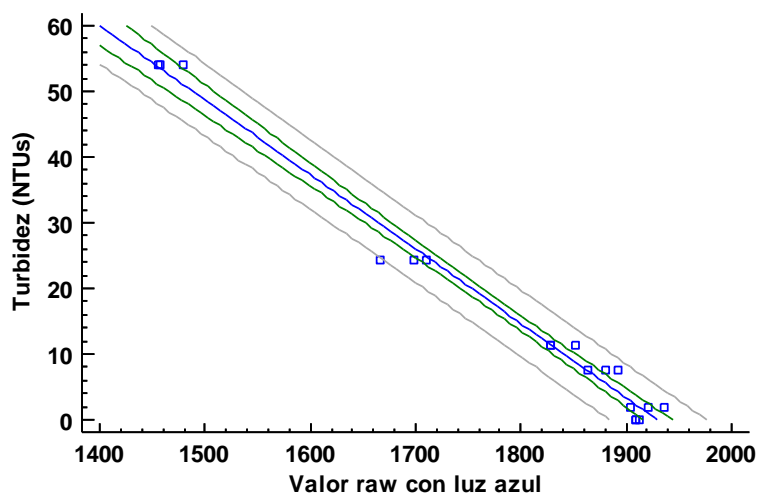


Figura 33 Recta de regresión de la materia vegetal en descomposición

$$Turbidez (NTUs) = 218.25 - 0.11 \times Raw \quad (3)$$

Por último, con la materia vegetal fresca separada por clases tal como podemos ver en la Figura 34. Obtenemos la recta de regresión de la materia vegetal fresca con la componente de luz azul obtenida en bruto. Posteriormente obtenemos la recta de regresión de la clase tal como se muestra en la Ecuación 4.

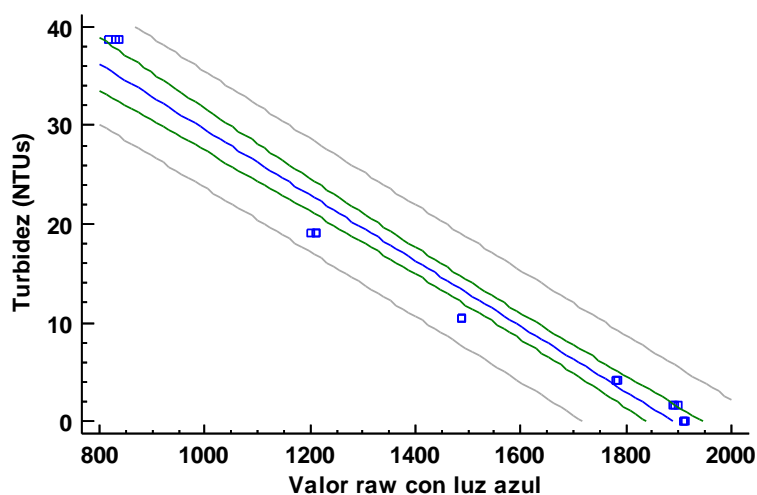


Figura 34 Recta de regresión de la materia vegetal fresca

$$Turbidez (NTUs) = 62.78 - 0.03 \times Raw \quad (4)$$

5.2 Medidas de vídeo en Lora

5.2.1 Tamaño de bloque

A partir de 256 bytes, se modificó el tamaño del bloque aumentando y disminuyendo el número máximo de bytes. Los valores probados oscilan entre 128 bytes y 6416 bytes. El MAT disminuye cuando aumenta el tamaño del bloque. El incremento del tamaño de bloque a valores bajos supone una gran reducción del MAT. Aumentar el tamaño del bloque de 4095 bytes a 6144 bytes supone una disminución del MAT de 2 s, lo que representa una disminución del 13% del MAT. Sin embargo, con un tamaño de 6400 bytes, los incrementos se reducen en MAT unos milisegundos y, después de 6416 bytes, falla el reenvío de vídeo. Todos estos datos se pueden ver en la Figura 35. La tendencia observada ha sido modelada matemáticamente, y el modelo y el coeficiente de correlación se incluyen en la Figura 35. Según los datos que obtuvimos, 6416 bytes es el tamaño de bloque óptimo para el resto de las pruebas. llevado a cabo en este escenario.

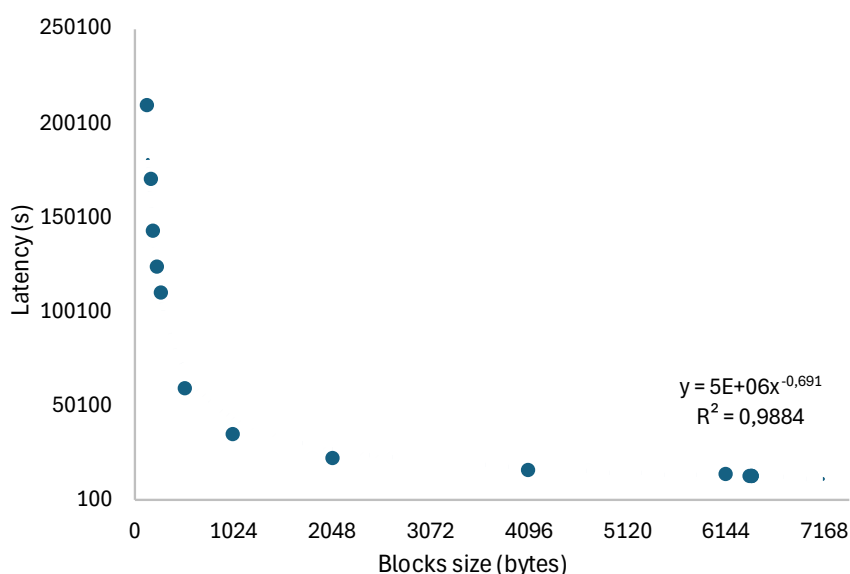


Figura 35 Gráfica MAT respecto al tamaño de bloque

5.2.2 Evaluación de la calidad máxima del video transmitido

Cabe destacar que en la prueba anterior el MAT mínimo alcanzado fue de 13,036 s, lo que supone casi el triple de la duración del vídeo. Por tanto, el siguiente paso es determinar qué calidad de vídeo se puede transmitir para lograr la transmisión de vídeo. Así, el vídeo original fue modificado para tener resoluciones de 256x144 píxeles, 426x240 píxeles y 640x360 píxeles. Además, el vídeo se ha enviado con y sin audio, ya que el audio no es estrictamente necesario en la mayoría de las aplicaciones de vigilancia. Los resultados se pueden ver en la Figura 36 y Figura 37. Comparamos el MAT para los diferentes videos y consideramos dos tamaños de bloque, el seleccionado en la primera prueba, 6408 bytes y 256 bytes. En la Figura 36 podemos ver que existe una clara correlación entre MAT y el tamaño del vídeo para cada uno de los tamaños de bloque utilizados. El MAT máximo para un tamaño de bloque igual a 256 bytes fue de 420,7 y

47,5 s para un tamaño de bloque igual a 6408 bytes. Mientras tanto, los valores mínimos son 60 y 7,6 s para tamaños de bloque de 256 y 6408 bytes.

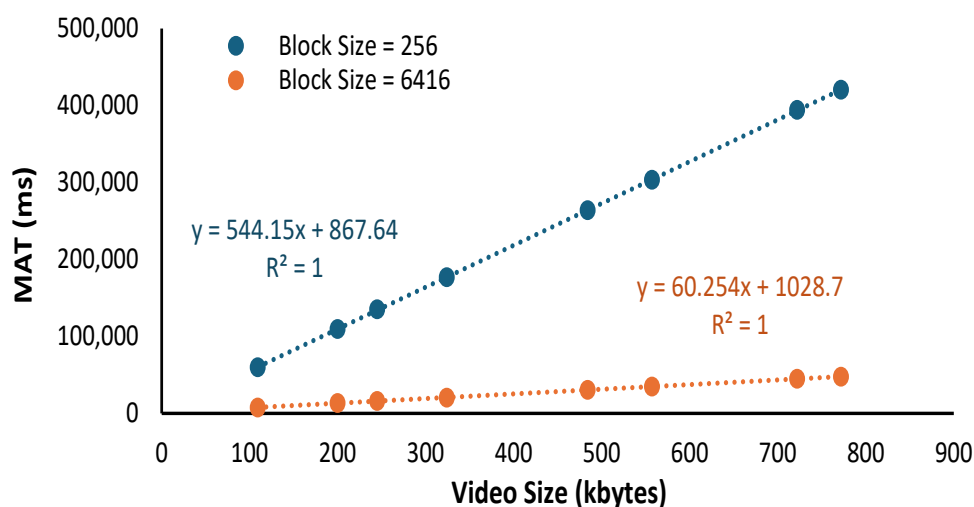


Figura 36 MAT en diferentes tamaños de video

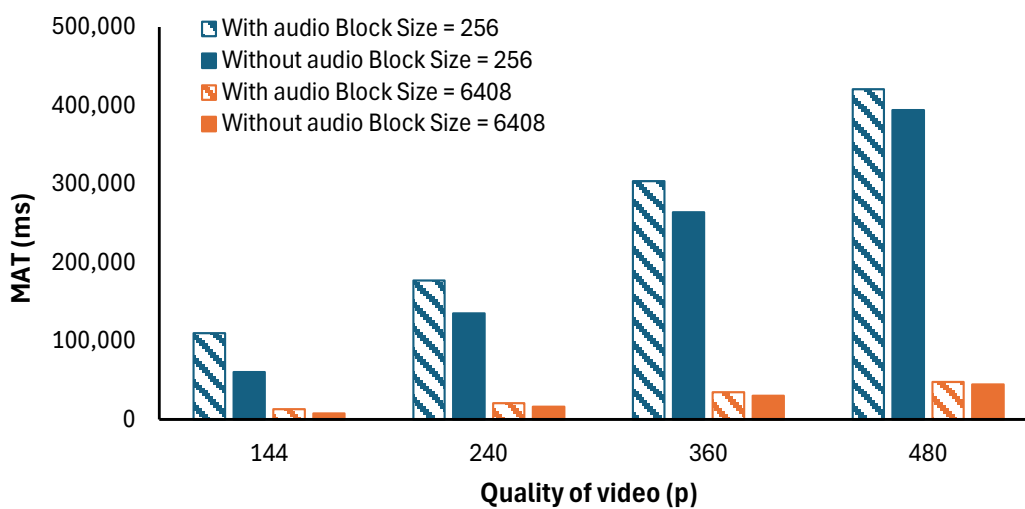


Figura 37 Variación del MAT en diferentes calidades de video

Respecto a la Figura 37, es posible ver el efecto de eliminar el audio en los videos con diferente calidad cuando se seleccionan diferentes tamaños de bloque. El impacto en el MAT es marginal cuando el tamaño del bloque es igual a 6408 bytes. Cuanto menor sea la resolución del vídeo, mayor será el porcentaje de reducción MAT obtenido cuando se elimina el audio.

5.2.3 Evaluación de MAT para diferentes tipos de extensiones de video

El siguiente parámetro estudiado es el efecto de la modificación del formato del vídeo y su impacto en el tamaño del vídeo y, por tanto, en el MAT. Entre las extensiones de vídeo probadas, .mkv es la que se caracteriza por un tamaño de vídeo menor, mientras que .avi tiene el tamaño máximo. En el caso de .mkv, el tamaño del vídeo es de 106 Kbytes cuando se incluye audio y de 51 Kbytes cuando se excluye el audio. En la Figura 38 representamos el MAT para las diferentes transmisiones de vídeo. El MAT en la mejor configuración fue de 4,1 s con .mkv sin audio, que fue similar al valor de .mov sin sonido (4,2 s). La posibilidad de incluir el audio propone un incremento de 3,4 y 5 s en el MAT para .mkv y .mov. El tiempo necesario para enviar el vídeo en formato avi con audio alcanza los 22,9 s con un tamaño de bloque de 6408 bytes y los 189 s con un tamaño de bloque de 256 bytes. El tamaño del vídeo y MAT de .mkv con audio es similar al de .mp4 sin audio. En esta prueba se ha conseguido tener un MAT inferior a la duración del vídeo, lo que indica la posibilidad de utilizar LoRa como tecnología para streaming de vídeo.

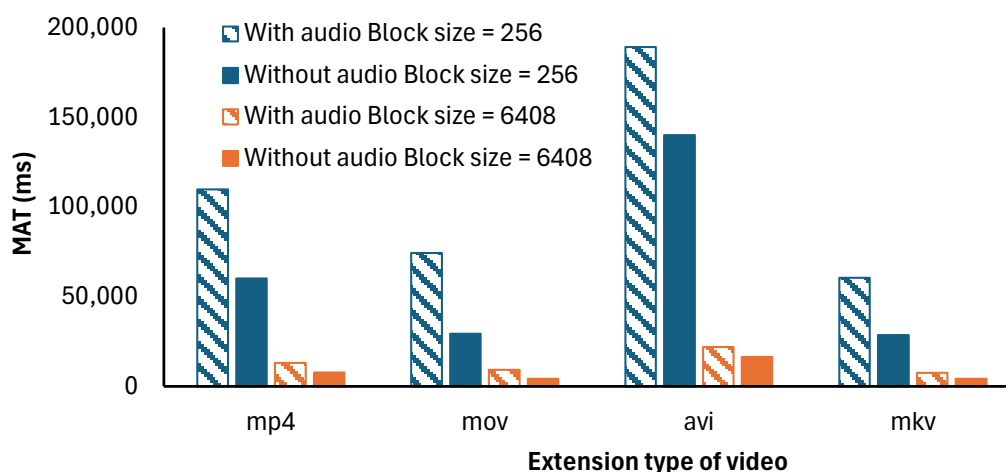


Figura 38 Variación del MAT para distintos tamaños de video

5.2.4 Desempeño del sistema propuesto en interiores.

En esta prueba, evaluamos el rendimiento de la transmisión de video en escenarios reales con nodos separados a diferentes distancias. Las pruebas iniciales con un tamaño de paquete igual a 6408 bytes arrojaron resultados poco claros y se perdieron muchos paquetes. Por tanto, se ha modificado el tamaño del bloque. Los datos del MAT obtenido para el tamaño de bloque probado se muestran en la Figura 39. La señal se recibe correctamente cuando el tamaño de bloque es menor que 6336 bytes. Como hemos determinado que el tamaño de bloque óptimo se ve afectado por la distancia, el resto de las pruebas se realizaron con un tamaño de bloque de 4095 bytes. Este valor equilibra un tamaño de bloque más pequeño al tiempo que garantiza que el MAT, 4,7 s, sea inferior a la duración del vídeo.

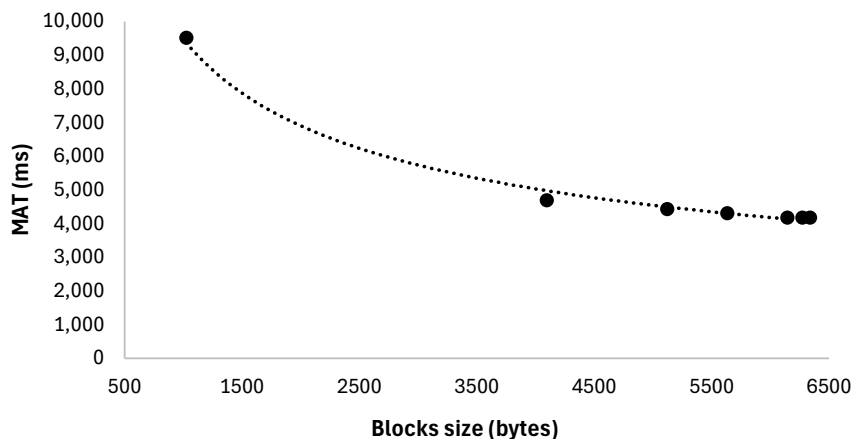


Figura 39 Efecto del tamaño de bloque al MAT

En el segundo lugar fue imposible establecer una buena conexión debido a múltiples obstáculos físicos. La tasa de pérdida de paquetes supera el 80%. Por lo tanto, podemos confirmar que los obstáculos (paredes) que se incluyeron afectan fuertemente la transmisión de video a través de la red LoRa.

5.2.5 Desempeño del sistema propuesto en exteriores.

Para la transmisión de vídeo en exteriores a largas distancias, y con el objetivo de mejorar los resultados, se ha reducido el bitrate del vídeo. El video usado fue .mp4 sin audio. Incluso en pruebas anteriores, el .mkv era el único formato que garantizaba la transmisión de vídeo; con la nueva tasa de bits, el tamaño se redujo considerablemente. El tamaño de los vídeos .mp4 después de la modificación de la tasa de bits es menor que el tamaño de los vídeos .mkv. Optamos por utilizar vídeo .mp4 ya que es uno de los formatos de vídeo estándar.

Las métricas utilizadas en este caso incluyen el RTT y la tasa de pérdida de paquetes. Dado que el entorno real sufre la variabilidad de posibles interferencias u obstáculos entre el emisor y el receptor, se realizaron tres repeticiones para cada transmisión de vídeo. Los resultados se presentan como el promedio y la desviación estándar de los datos. Los resultados del RTT, como se ve en la Figura 40, indican que la transmisión de video fue posible para videos de 256x144 y 426x240 con un RTT promedio de 2,1 y 3,2 s para el video con una duración de 5 y sin audio. Mientras tanto, para vídeo con 854x480, el RTT promedio fue de 10,3 s. Para el vídeo con 640x360 hubo una medida anormal en la segunda transmisión, con RTT igual a 113 s. El RTT promedio de las otras dos transmisiones fue de 8,9 s.

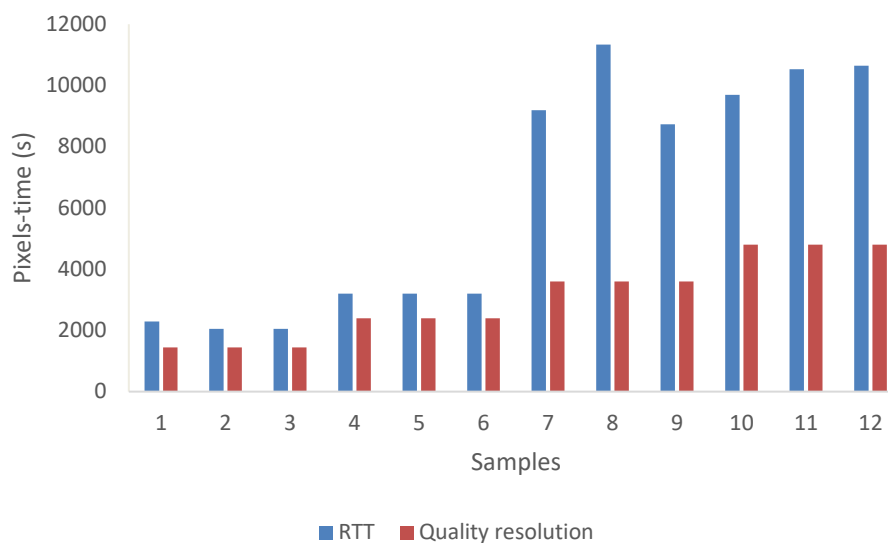


Figura 40 RTT respecto a la calidad del video a distancia

Esta es una foto de donde se mandaron los videos a unos de los edificios de la playa como se puede ver en la Figura 41.



Figura 41 Visual de donde se envió el video en exteriores

5.3 Configuración de los nodos para la transmisión Lora

5.3.1 Medidas de distancia en Lora

Para la transmisión en Lora mi primera duda era saber a cuanto distancia era capaz de transmitir. Para ello calcule haciendo un ping (tiempo de transmisión al medio + 2 * tiempo de propagación + 2 * tiempo de procesamiento) viendo si variaba a distintas distancias y si el tiempo aumentaba al paso de que aumentaran la distancia entre los nodos viendo que desde la distancia máxima que son aproximadamente 15000 metros hasta la mínima de unos 500 metros el tiempo entre los pines era el mismo tal como se puede ver en la Figura 42.

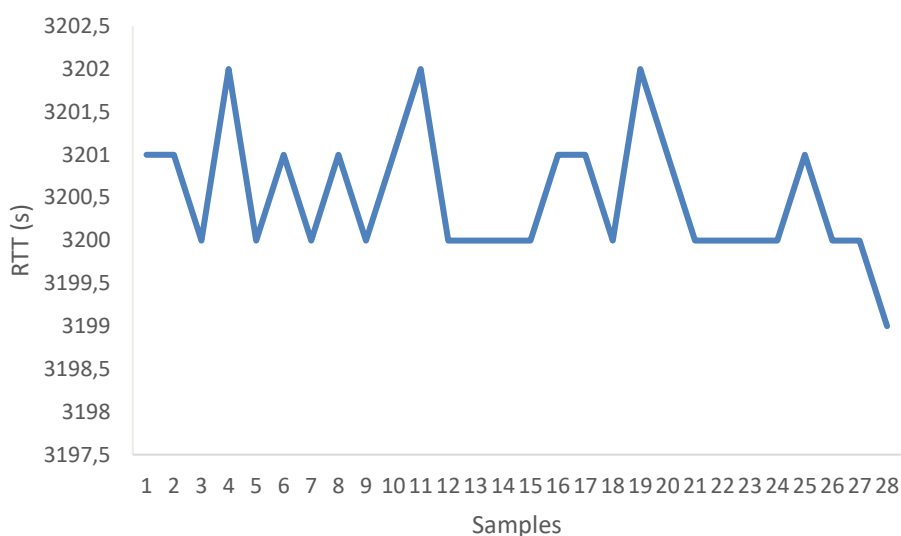


Figura 42 RTT haciendo un ping en Lora

Todo ello situado en lo alto de las colinas de mi pueblo para que pueda haber línea directa entre los dispositivos. El experimento está incompleto ya que quedara ver con líneas de transmisión directa cual es la distancia máxima a la que se podría hacer el ping.

Por último, en la Figura 43 podemos ver las distancias y las distintas localizaciones utilizadas para la captura de medidas y datos en Lora durante la elaboración del proyecto.

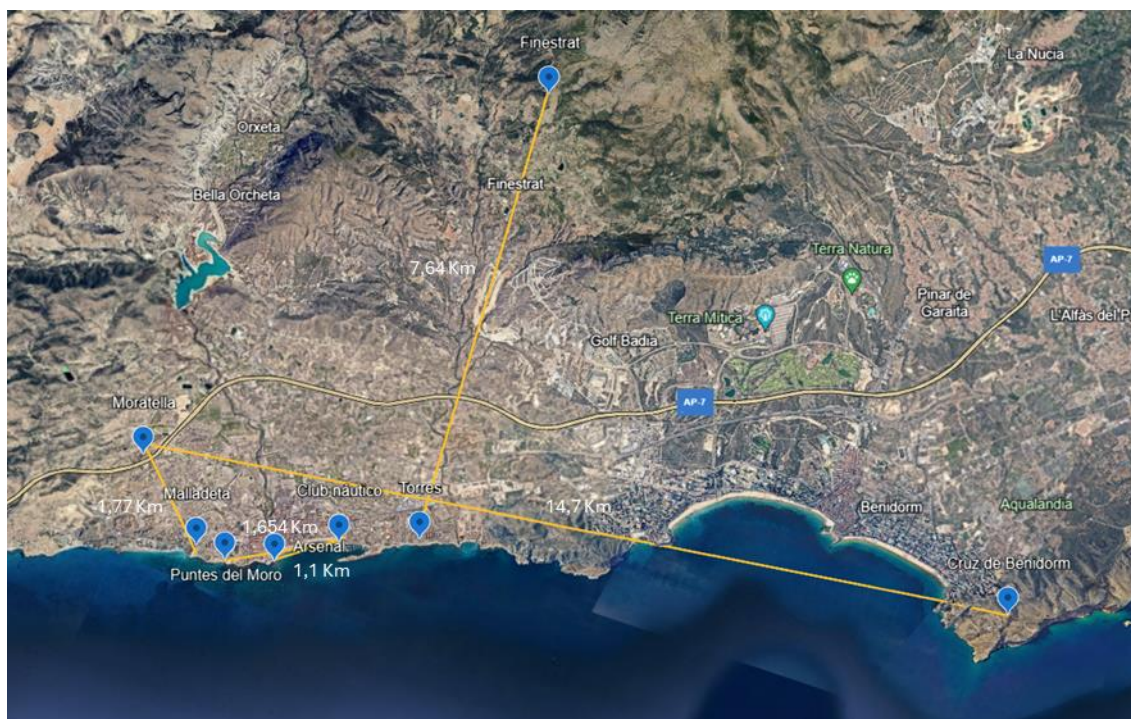


Figura 43 Mapa de las localizaciones y distancias donde se han hecho las medidas



Capítulo 6. Conclusiones

6.1 Cumplimiento del objetivo

Durante el desarrollo del proyecto se ha logrado desarrollar, calibrar y aislar sensores de bajo coste para conductividad, turbidez y temperatura tal como se puede ver en el capítulo de desarrollo y de pruebas. Se han probado y nos han dado muy buenos resultados en la puesta de escena en el laboratorio y han funcionado satisfactoriamente. Con respecto al objetivo de proponer y testear un sistema de comunicación basado en Lora que permita el envío de vídeo el cual detallado en el apartado de la propuesta y las pruebas hemos visto. Por último, tal como hemos visto en el capítulo de la propuesta en la parte del cliente web con base de datos y conectado al servidor se ha cumplido el objetivo de hacer el diseño de una base de datos para acceso a los datos tomados por el nodo.

6.2 Conclusiones sobre el proyecto

La posibilidad de dar una alternativa a los observatorios de las variables marítimos-costeras de bajo coste a la red del litoral español ha sido muy satisfactorio. Una de las cosas más punteras del proyecto es la combinación de distintas tecnologías como pueden ser las comunicaciones con Lora o Ethernet junto al uso del internet de las cosas con una web para monitorizar los datos brinda un proyecto de un uso fluido de la información que se obtiene de los sensores.

También cabe destacar el uso de electrónica analógica con la implementación de circuitos impresos para los sensores creados de elaboración propia y como se deben de adaptar al encapsulado. Respecto a los componentes de la demo en el ámbito de cómo se han adaptado para absorber las exigencias técnicas de mezclar la electrónica con el agua se ha superado correctamente.

A lo largo del proyecto, se han aplicado conocimientos adquiridos durante el grado, especialmente en áreas como:

- Programación: Mediante asignaturas como "Programación 1" y "Programación 2", donde se adquirieron las bases fundamentales de la programación.
- Redes de área local: A través de la asignatura "Redes de área local", que proporcionó el conocimiento necesario para las comunicaciones con Lora y comunicaciones IoT.
- Proyecto Internet de las cosas y Aplicaciones móviles: Gracias a la asignatura "proyecto Internet de las cosas y Aplicaciones móviles", que enseñó cómo diseñar sistemas IoT con dispositivos finales para el cliente.
- Proyecto diseño y programación Web: Gracias a la asignatura proyecto diseño y programación Web que enseñó cómo diseñar e implementar clientes web.

Por último, la posibilidad de crear el observatorio me ha sumergido en el tema de la investigación. El cual me apasiona el poder llevar a cabo un proyecto con aportaciones novedosas a la ciencia con la elaboración de artículos como pueden ser del calibrado de la bobina del sensor de conductividad para posteriormente integrar el modelo en Arduino o el estudio de una red Lora de comunicación punto a punto para streaming de video han sido el inicio de un camino que me gustaría llevar los próximos años.

6.3 Problemas encontrados y cómo se han solucionado

En esta subsección me gustaría destacar la problemática que ha habido con alimentar correctamente con el generador de señales las bobinas del sensor de conductividad. El problema yace en que los sensores pueden variar sus características en las bobinas dependiendo el montaje del circuito siendo muy susceptible a los campos. Teniendo al principio que hacer siempre que lo volvía a utilizar un barrido de frecuencias para volver a encontrar la frecuencia de resonancia de la bobina siendo esto muy laborioso y pesado.

Como es obvio cuando se utiliza proyectos donde se integran códigos de programación la resolución de errores típicos de sintaxis o de operación de los compiladores o interpretadores de los lenguajes que he utilizado. Como error de compilación destacado sería el de la política de CORS cuando configure el servidor HTTP en local para lanzar un servicio DNS para poder acceder a la web desde fuera de la red.

6.4 Aportaciones personales

Para mí la realización de este proyecto me ha llenado mucho. Desde pequeño siempre me ha gustado ingeniar cosas y realizar investigaciones y productos con cosas tecnológicas. Y el poder llevar a cabo un proyecto de esta envergadura ha sido el broche final a una carrera que me ha llenado de conocimientos y ha cumplido con crecer los objetivos que tenía cuando la empecé.

Poder llevar a cabo un proyecto como un producto tecnológico desde el encapsulado y el hardware con las comunicaciones y todo el software del microcontrolador hasta la parte del cliente web me llena de orgullo porque era gusto las habilidades que quería adquirir cuando empecé la carrera y desde que era pequeño.

6.5 Publicaciones

Respecto a las publicaciones realizadas se han hecho dos artículos de congreso. El primero es el de una red Lora punto a punto para el envío de streaming de video tal como se ha detallado en esta memoria. Otra es la calibración del sensor de conductividad para crear un modelo para poder integrarlo en Arduino. Y por último es el análisis de un sistema IoT el diseño y la arquitectura propuestas. Aquí está la lista de los artículos:

- Ivars-Palomares, A., Zaragoza-Esquerdo, M., Sendra, S., & Lloret, J. Design of a platform for data storage and data management in IoT networks 10th International Conference on Wireless Networks and Mobile Communications (WINCOM), Istanbul, Turkey, October 26 -28, 2023.
- Miguel Zaragoza-Esquerdo, Vinie Lee Silva-Alvarado, Lorena Parra, Sandra Sendra, Jaime Lloret and Manuel Pulido-Velázquez. Performance of ML-based Classification Models as Edge Computing in IoT Nodes for a Marine Observatory. 3rd Eurasian Conference on Frontiers of Computer Science and Information Technology (FCSIT), Barcelona, Spain, September 20-22, 2024
- Miguel Zaragoza-Esquerdo, Ms. Lorena Parra, Dr. Sandra Sendra and Prof. Jaime Lloret *LoRa Video Streaming in Rural Wireless Multimedia Sensor Networks*. 19th International Symposium on Wireless Communication Systems (ISWCS), Rio de Janeiro, Brazil, July 14-17, 2024



- Miguel Zaragoza-Esquerdo, Alberto Ivars-Palomares, Sandra Sendra, Jaime Lloret
Diseño de una Plataforma para el Almacenamiento y Gestión de los datos en Redes IoT
XVI jornadas de Ingeniería telemática (JITEL), Barcelona, España, noviembre 8-10, 2023

6.6 Futuras líneas de trabajo

Después de la realización de este trabajo de final de grado se podría implementar el uso de inteligencia artificial. Como por ejemplo redes neuronales artificiales utilizando los sensores del nodo multiparamétrico para crear un sensor virtual que detecte diferentes compuestos líquidos que se encuentren en las muestras que se están midiendo.

Con ello se podrán detectar compuestos contaminantes que se encuentren en el agua como pueden ser vertidos de aguas fecales o de las petroleras. Por lo tanto, se podrá llevar un control medioambiental del océano con un sistema de alertas en tiempo real.

Por último, otra investigación que se va a hacer es medir la eficacia de transmitir comunicaciones Lora debajo del agua viendo su comportamiento (paquetes perdidos, RTT, etc.) en este medio mediante la adaptación de los nodos emisor y receptor en el medio marino.



Capítulo 7. Referencias

- [1] Zoido Naranjo, F., & Arroyo Pérez, A. (2003). La población de España. *Tendencias demográficas durante el siglo XX en España*.
- [2] Ivars-Palomares, A., Zaragoza-Esquerdo, M., Sendra, S., & Lloret, J. Design of a platform for data storage and data management in IoT networks 10th International Conference on Wireless Networks and Mobile Communications (WINCOM), Istanbul, Turkey, October 26 -28, 2023.
- [3] Sendra, S., Ivars-Palomares, A., Zaragoza-Esquerdo, M., & Lloret, J. Group-Based Collaborative Environments for Coastal Areas Monitoring. The 20th International Conference on Cooperative Design, Visualization and Engineering (CDVE). Mallorca, Spain, October 1-4, 2023.
- [4] Viciano-Tudela, S., Parra, L., Sendra, S., & Lloret, J. (2023). A low-cost virtual sensor for underwater ph monitoring in coastal waters. *Chemosensors*, 11(4), 215.
- [5] Parra, L., Viciano-Tudela, S., Carrasco, D., Sendra, S., & Lloret, J. (2023). Low-cost microcontroller-based multiparametric probe for coastal area monitoring. *Sensors*, 23(4), 1871.
- [6] Sendra, S., Parra, L., Jimenez, J. M., Garcia, L., & Lloret, J. (2023). LoRa-based network for water quality monitoring in coastal areas. *Mobile Networks and Applications*, 28(1), 65-81.
- [7] Sendra, S., Viciano-Tudela, S., Ivars-Palomares, A., & Lloret, J. Low-Cost Water Conductivity Sensor Based on a Parallel Plate Capacitor for Precision Agriculture. In *International Conference on Advanced Intelligent Systems for Sustainable Development (AI2SD)*. Rabat, Morocco, May 22-27, 2022.
- [8] Parra, L., Sendra, S., Lloret, J., & Rodrigues, J. J. Low-cost wireless sensor network for salinity monitoring in mangrove forests. In *IEEE SENSORS, Valencia, Spain, November 2-5, 2014*.
- [9] Lloret, J., Sendra, S., Ardid, M., & Rodrigues, J. J. (2012). Underwater wireless sensor communications in the 2.4 GHz ISM frequency band. *Sensors*, 12(4), 4237-4264.
- [10] Sendra, S., Lloret, J., Rodrigues, J. J., & Aguiar, J. M. (2013). Underwater wireless communications in freshwater at 2.4 GHz. *IEEE Communications Letters*, 17(9), 1794-1797.
- [11] Soto Pérez, M. (2022). Diseño de un sistema de monitorización de la calidad del agua para zonas costeras. Universitat Politècnica de València. <http://hdl.handle.net/10251/188200>
- [12] Gómez Fernández-Pacheco, J. (2022). Desarrollo de un sistema de sensores para la monitorización de Posidonia Oceánica. Universitat Politècnica de València. <https://m.riunet.upv.es/bitstream/handle/10251/189177/Gomez%20-%20Desarrollo%20de%20un%20sistema%20de%20sensores%20para%20la%20monit%20orizacion%20de%20Posidonia%20Oceanica%20L%20Delil....pdf?sequence=1&isAllowed=y>



- [13] Das, B.; Jain, P.C. Real-time water quality monitoring system using Internet of Things. International Conference on Computing, Communications and Electronics (Comptelix), Jaipur, India, July 1-2, 2017.
- [14] Parra, L.; Sendra, S.; Lloret, J.; Bosch, I. Development of a conductivity sensor for monitoring underground water resources to optimize water management in smart city environments. In IEEE SENSORS, Busan, Corea, November 1-4, 2015.
- [15] Wang, Y., Liu, A., Han, Y., & Li, T. (2020). Sensors based on conductive polymers and their composites: a review. *Polymer International*
- [16] Azman, A.A.; Rahiman MH, F.; Taib, M.N.; Sidek, Nuevo Hampshire; Bakar IA, A.; Ali, M. F. A low cost nephelometric turbidity sensor for continual domestic water quality monitoring system. In Proceedings of the 2016 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Selangor, Malasia, October 22, 2016.
- [17] Arifin, A.; Irwan, I.; Abdullah, B.; Tahir, D. Design of water turbidity sensors based on polymeric fiber optics. In Proceedings of the International Seminar on Sensors, Instrumentation, Measurement and Metrology (ISSIMM), Surabaya, Indonesia, August 25-26, 2017.
- [18] Wang, Y.; Rajib SS, M.; Collins, C.; Grieve, B. Low-cost turbidity sensor for low-power wireless monitoring of freshwater courses. *Sensores IEEE*, New Delhi, India, October 28-31, 2018.
- [19] Mulyana, Y.; Hakim, D.L. Prototype of water turbidity monitoring system. International Symposium on Materials and Electrical Engineering (ISMEE), Bandung, Indonesia, November 16, 2017.
- [20] Parra, L.; Rocher, J.; Escrivá, J.; Lloret, J. Design and development of a low-cost intelligent turbidity sensor for monitoring water quality in fish farms. *Aquac. Eng.* 2018, 81, 10–18.
- [21] Mansor, H.; Maju NA, H.; Gunawan, T.S.; Ahmad, R. The development of a water contamination detector using the principles of conductivity and turbidity. *International Islamic University of Malaysia. J.* 2022, 23, 104–113.
- [22] Sharaf El Din, E.; Zhang, Y.; Suliman, A. Mapping surface water quality parameter concentrations using a new remote sensing and artificial intelligence framework. *Int. J. Remote Sens.* 2017, 38, 1023–1042.
- [23] Abdelmalik, K.W. Role of statistical remote sensing for the prediction of quality parameters of continental waters. *Egypt. J. Remote Sens. Space Sci.* 2018, 21, 193–200.
- [24] Sagan, V.; Peterson, K. T.; Maimaitijiang, M.; Sidike, P.; Sloan, J.; Greeling, B.A.; Maalouf, S.; Adams, C. Monitoring Inland Water Quality Using Remote Sensing: Potential and Limitations of Spectral Indices, Bio-Optical Simulations, Machine Learning, and Cloud Computing. 2020, 205, 103187.



- [25] Agosto de 2018, [en línea] Disponible: <https://www.semtech.com/technology/lora>.
- [26] LoRa Documentation. Available online: <https://lora.readthedocs.io/en/latest/> (accessed on 1 February 2020).
- [27] Froiz-Míguez, I.; Fraga-Lamas, P.; Varela-Barbeito, J.; Fernández-Caramés, T.M. LoRAWAN and Blockchain based Safety and Health Monitoring System for Industry 4.0 Operators. *Proceedings 2019*, 6, 1–6.
- [28] Sanchez-Iborra, R.; Sanchez-Gomez, J.; Ballesta-Viñas, J.; Cano, M.D.; Skarmeta, A.F. Performance evaluation of LoRa considering scenario conditions. *Sensors* 2018, 18, 772.
- [29] Benites, B.; Chávez, E.; Medina, J.; Vidal, R.; Chauca, M. LoRaWAN applied in Swarm Drones: A focus on the use of fog for the management of water resources in Lima-Peru. In *Proceedings of the 5th International Conference on Mechatronics and Robotics Engineering*, Rome, Italy, 16–19 February 2019; pp. 171–176.
- [30] Sisinni, E.; Ferrari, P.; Fernandes Carvalho, D.; Rinaldi, S.; Pasetti, M.; Flammini, A.; Depari, A. A LoRaWAN range extender for Industrial IoT. *IEEE Trans. Ind. Inform.* 2019, 1–10.
- [31] Jia Y (2020) LoRa-Based WSNs Construction and Low-Power Data Collection Strategy for Wetland Environmental Monitoring. *Wireless Pers Commun* 114:1533–1555
- [32] Yan-Ting L, Bo-Yi L, Xiao-Feng Y, Zong-Xuan C, Zi-Xian Y, Wei-Hong L, Song-Yi H, Jun-Lin L, Jing-Wen P, Jen-Yeu C (2018) A Solar powered long range real-time water quality monitoring system by LoRaWAN. 2018 27th Wireless and Optical Communication Conference (WOCC). Hualien, Taiwan
- [33] Das B, Jain PC (2017) Real-time water quality monitoring system using Internet of Things. 2017 Int. Conf. on Computer, Communications and Electronics (Comptelix). Jaipur, India
- [34] Simitha KM, Subodh Raj MS (2019) IoT and WSN based water quality monitoring system. In *proc. of 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, June 2–14, 2019, Coimbatore, India, pp. 205–210
- [35] Wu N, Khan M (2019) LoRa-based internet-of-things: a water quality monitoring system. In: 2019 SoutheastCon, April 11–14, 2019, Huntsville, Alabama-USA, pp. 1–4
- [36] ALLDATASHEET, Disponible en: https://www.alldatasheet.com/view.jsp?Searchword=Ds18b20%20datasheet&gad_source=1&gclid=CjwKCAjw1920BhA3EiwAJT3ISUMrJnLLCpOCbEd9hUIZieWknOocx8eobEHwrcQ3UOrD2riLKsyAtRoCBbMQAvD_BwE [Ultimo acceso: 04/07/2024]