

GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA

TRABAJO DE FIN DE GRADO

Red privada LoRaWAN para el ámbito de aplicaciones en la

Industria 4.0

Autor: Daniel Rodríguez Díaz

Tutor: José Antonio Soria Pérez

28 de septiembre de 2021

RESUMEN

En este proyecto se ha realizado el montaje y configuración de una red privada LoRaWAN (Long Range Wide Area Network), imitando la estructura que esta debería tener para ser implementada en la industria 4.0.

Este tiene como principal objetivo establecer los pasos y procedimientos básicos para la implementación de la tecnología LoRa en el ámbito industrial. También estudia la situación actual de la tecnología, analizando su grado de implementación en diferentes países y sus ejemplos de uso.

El prototipo está formado por tres dispositivos sensores que incorporan tecnología LoRa, una puerta de enlace (gateway), un servidor de red (Network Server) LoRaWAN para gestionar los tres sensores, una aplicación Node-RED, un servidor OPC (Open Platform Communication) y una aplicación SCADA (Supervisory Control And Data Acquisition) de demostración. En esta última, se simula un sistema que monitoriza los sensores situados en un contenedor refrigerado de fruta.

Gracias a la fácil modularidad de la plataforma escogida, esta permite escalar fácilmente los proyectos ya que se pueden incorporar o quitar dispositivos de manera modular, pudiendo adaptar cualquier sistema de producción diseñado en planta. Simplemente aumentando o disminuyendo el número de dispositivos y variando su modelo y función, se puede llegar a obtener infinitas instancias de red privada LoRaWAN.

Con todo esto, se concluye que la implementación de redes privadas LoRaWAN es una elección de rápido diseño y montaje, con gran utilidad en el establecimiento de comunicaciones no críticas en procesos industriales.

Índice general

1. Introducción	9
1.1. Antecedentes	9
1.2. Objetivo	10
1.3. Alcance	10
1.4. Requerimientos	11
1.5. Justificación	13
2. Estado del arte	14
2.1. LoRaWAN y la especificación LoRa	14
2.1.1. Contexto y definición	14
2.1.2. Características	15
2.2. Estructura de las redes LoRaWAN	16
2.2.1. Nodos	18
2.2.2. Gateway	21
2.2.3. Servidor red	22
2.2.4. Servidor de aplicaciones	22
2.2.5. Servidor de unión	23
2.3. Aplicaciones de LoRaWAN	23
2.3.1. Contexto Internet of Things	23
2.3.2. Industrial Internet of Things	23
2.3.3. Comparación frente a otras tecnologías inalámbricas	24
2.3.4. Ejemplos de uso	27
2.4. Protocolos de comunicación	33
2.4.1. MQTT	33
2.4.2. OPC UA	34

3. Planteamiento y elección de recursos	38
3.1. Perspectiva general	38
3.2. Hardware	39
3.2.1. Nodos utilizados	39
3.2.2. Gateway	45
3.3. Software	47
3.3.1. Servidor red ThingPark Enterprise	47
3.3.2. Node-RED	50
3.3.3. Servidor OPC KEPServerEX	51
3.3.4. Cliente SCADA	53
4. Desarrollo de la red	57
4.1. Instalación física de los equipos	57
4.1.1. Instalación del Gateway	57
4.1.2. Instalación del sensor de temperatura y humedad	58
4.1.3. Instalación del sensor de fuga de agua	59
4.1.4. Instalación del sensor de nivel	59
4.2. Configuración en ThingPark Enterprise	61
4.2.1. Configuración del Gateway	61
4.2.2. Dada de alta del Gateway en ThingPark Enterprise	64
4.2.3. Creación del conector ThingPark Enterprise - Broker MQTT de Mosquitto	64
4.2.4. Dada de alta de los sensores en ThingPark Enterprise	67
4.3. Acondicionamiento con Node-RED	67
4.3.1. Flujo para el sensor de temperatura y humedad	68
4.3.2. Flujo para el sensor de fuga de agua	70
4.3.3. Flujo para el sensor de nivel	70
4.4. Creación del proyecto de KEPServerEX	71
4.4.1. Suscripción a los topics del Broker	71
4.4.2. Creación de tags para el envío de downlinks	73
4.4.3. Creación del cliente MQTT publicador	74
4.5. Desarrollo del SCADA en AVEVA Edge	76
4.5.1. Creación de los tags de proyecto y comunicación con el servidor OPC	76
4.5.2. Creación y diseño de pantallas	77

5. Resultados de la plataforma	86
6. Resumen del presupuesto	88
7. Conclusiones	90
A. Presupuesto	98

Índice de figuras

2.1. Comunicación mediante LoRaWAN [12]	14
2.2. Estructura estándar de las redes LoRaWAN [18]	17
2.3. Ejemplo de nodo LoRaWAN [21]	18
2.4. Ejemplo de puerta de enlace LoRaWAN [28]	21
2.5. Usos comunes de LoRaWAN [41]	28
2.6. Disposición de los transmisores LoRa [43]	30
2.7. ACW-DIND88 [45]	32
2.8. Ejemplo de arquitectura MQTT [48]	34
2.9. Diferencia de plataformas entre OPC DA y OPC UA [51]	35
2.10. Arquitectura Cliente - Servidor [52]	36
2.11. Intercambio de certificados entre cliente y servidor [54]	37
3.1. Esquema de la red prototipo	40
3.2. Vista del nodo ACW/LW8-TH [55]	40
3.3. Vista del nodo ACW/LW8-WL [56]	42
3.4. Vista del nodo ACW/LW8-LVL [57]	43
3.5. Diferentes modos del Gateway ODU Macro [6]	45
3.6. Esquema de conexionado del Gateway ODU Macro [58]	46
3.7. Puertos físicos del Gateway ODU Macro [59]	46
3.8. Interfaz de ThingPark Enterprise [60]	48
3.9. ThingPark Enterprise en una red LoRaWAN estándar [7]	49
3.10. Interfaz Túnel entre ThingPark Enterprise y los Application Servers [62]	50
3.11. Interfaz de Node-RED [63]	51
3.12. Comunicaciones establecidas por KEPServerEX [9]	52
3.13. Interfaz de KEPServerEX Configuration	53

3.14. Propiedades del proyecto de KEPServerEX	54
3.15. KEPServerEX OPC UA Configuration Manager	54
3.16. Interfaz de Aveva Edge 2020 Studio	55
4.1. Esquema acotado del nodo ACW/LW8-TH [3]	58
4.2. Placa interior del nodo ACW/LW8-TH [3]	58
4.3. Imán de activación del nodo ACW/LW8-WL [5]	60
4.4. Esquema acotado del nodo ACW/LW8-WL [5]	60
4.5. Rastreo de IPs en la red mediante Advanced IP Scanner	61
4.6. Conexión SSH mediante Putty	62
4.7. Menú principal del software LRR del Gateway	62
4.8. LRR UID del Gateway	63
4.9. Menú VPN Configuration del software LRR	63
4.10. Public key generada en el Gateway	63
4.11. Configuración de los parámetros PKI del gateway	64
4.12. Creación de un nuevo gateway en ThingPark Enterprise	64
4.13. Establecimiento de parámetros del gateway en ThingPark Enterprise	65
4.14. Indicadores de estado del gateway en ThingPark Enterprise	65
4.15. Tipos de salidas de datos en ThingPark Enterprise	66
4.16. Parámetros del conector MQTT en ThingPark Enterprise	66
4.17. Creación de un dispositivo LoRa genérico en ThingPark Enterprise	67
4.18. Flujos de acondicionamiento del sensor TH	68
4.19. Flujo de acondicionamiento del sensor de fuga de agua	70
4.20. Flujo de acondicionamiento del sensor de nivel	71
4.21. Configuración del canal MQTT en KEPServerEX	71
4.22. Canales y dispositivos creados en el proyecto de KEPServerEX	72
4.23. Acceso al topic del sensor TH desde KEPServerEX	72
4.24. Suscripción de cada dispositivo a los topics del broker	73
4.25. Dispositivos creados para el envío de downlinks	73
4.26. Creación de los tags para el envío de downlinks	73
4.27. Propiedades del cliente MQTT publicador en KEPServerEX	74
4.28. Selección de la plantilla avanzada de mensaje JSON	74
4.29. Formato de downlink compatible con ThingPark Enterprise [66]	75

4.30. Ítem de asignación al tag a publicar	75
4.31. Hoja de datos del proyecto en AVEVA Edge	76
4.32. Propiedades de la conexión OPC UA con KEPServerEX	76
4.33. Conexión de tags entre AVEVA Edge y KEPServerEX	77
4.34. Configuración del proyecto de AVEVA Edge	78
4.35. Pantallas del proyecto de AVEVA Edge	78
4.36. Interfaz de la pantalla principal del proyecto	79
4.37. Propiedades del botón 'Vista de la planta'	79
4.38. Asignación del cuadro de texto al valor de temperatura	80
4.39. Animación de visibilidad para alarma de temperatura	80
4.40. Interfaz de la pantalla 'Vista Planta'	82
4.41. Animación de posición de la compuerta inferior	83
4.42. Interfaz de la pantalla 'Histórico de valores'	84
4.43. Configuración de los ejes del histórico	85
5.1. Nodos debug en los flujos	86
5.2. Salida del flujo para el acondicionamiento de la temperatura	87
5.3. Visualización de las variables en Quick Client	87
5.4. Visualización de las variables en el editor de AVEVA Edge 2020	87

Índice de cuadros

2.1. Tabla comparativa de las tecnologías de redes LPWAN	27
3.1. Formatos de payload del sensor T/H [3]	41
3.2. Formato de downlink de periodo de muestreo [3]	42
3.3. Byte 1 del uplink de lectura del ACW/LW8-LVL [5]	44
3.4. Byte 2 del uplink de lectura del ACW/LW8-LVL [5]	44
3.5. Bytes de distancia medida del ACW/LW8-LVL [5]	44
6.1. Resumen del coste de la implementación de la red desarrollada	89
A.1. Coste del hardware y su instalación	99
A.2. Coste del software	99
A.3. Coste del personal	99
A.4. Suma del coste del hardware, software y personal	100

Capítulo 1

Introducción

1.1. Antecedentes

Actualmente el concepto de IoT (Internet of Things) e Industria 4.0 se aplica ampliamente en el desarrollo de redes de sensores que tienen la capacidad de enviar información sin requerir intervención humana. En el ámbito industrial, ciertas aplicaciones requieren de redes que puedan enviar datos entre estaciones de trabajo lejanas y de manera inalámbrica. En este sentido, LoRaWAN es un protocolo de red que usa la tecnología LoRa, para redes de baja potencia y área amplia que en el ámbito industrial apenas ha llegado a desplegarse. Esto en parte se debe a que un gran número de las comunicaciones en este sector son críticas, y sus necesidades de cantidad y velocidad de transmisión de datos no pueden ser cubiertas con las características de LoRa [1].

Especialmente en España, LoRaWAN está presente en sectores como Smart Cities y e-agriculture. Sin embargo, su implantación en la industria se encuentra en una fase muy preliminar, tanto por desconocimiento como por falta de opciones e infraestructura. Por ello, con el montaje de esta red se ofrecería una solución completa, haciendo fácil la integración de la tecnología y los dispositivos LoRa con elementos y funcionalidades ya consolidadas en las empresas de este país. Se pretende demostrar la adaptabilidad y el potencial de la tecnología para funciones específicas dentro de las plantas industriales [2].

La principal ventaja de este enfoque es su escalabilidad. La red del proyecto es finita, se utilizarán algunos dispositivos concretos y se explorarán ciertas funcionalidades con ellos. Sin embargo, el hecho de detallar y dejar constancia de todo el proceso seguido hace que sea posible su modificación y expansión para los requisitos propios de cada cliente. Según la funcionalidad buscada o los recursos materiales de los que se disponga esta guía le permite al cliente configurar su red privada según sus necesidades.

Como principal desventaja cabe destacar la imposibilidad de testear su verdadero potencial. Debido a las limitaciones de movilidad y presencialidad, no es posible realizar este montaje en una planta real. En

consecuencia, los dispositivos no medirán datos relevantes para procesos industriales.

Ya que el prototipo se realiza en una vivienda, también es difícil comprobar el potencial que posee LoRa para transmitir datos a largas distancias.

1.2. Objetivo

El presente trabajo de fin de estudios tiene como objetivo la creación de una red privada LoRaWAN para su posible aplicación en la industria. Con ello se quiere lograr tener una solución completa que sirva como punto de partida para cualquier empresa que esté pensando incorporar tecnología LoRa en su planta. Para facilitar su implementación, se pretende acompañar con softwares y hardwares ya estandarizados en el sector. Esto permitirá que la red no sea un proceso separado e independiente, sino que sea una adición más a procesos industriales ya existentes.

También se pretende analizar el estado de la tecnología LoRa y las redes LoRaWAN en el mundo. Esto incluye su despliegue actual, sus casos más habituales de uso y ejemplos de su implementación en el ámbito industrial.

1.3. Alcance

Para llegar a cumplir los objetivos marcados del presente proyecto, primeramente será necesario comprender la tecnología en la que se va a basar. Por ello, debe realizarse un estudio teórico de LoRa y las redes LoRaWAN; funcionamiento, estructura, clasificaciones dispositivos.

Con el objetivo de valorar su aplicación en la industria, también será vital conocer el contexto de las redes LoRaWAN, viendo la concepción que se tiene sobre ellas y casos reales de implementación.

Finalmente, también es necesario comprender el funcionamiento de el software y hardware que va a conformar la red. Se deberá estudiar la documentación de los dispositivos disponibles, así como aprender a utilizar el programario encargado de las comunicaciones y la monitorización de la información.

A continuación se enumeran las tareas que denotan el alcance del proyecto:

- Estudio de la tecnología LoRa y las redes privadas LoRaWAN.
- Análisis de el estado actual de LoRa, en especial en el ámbito industrial.
- Estudio de la documentación referente a los dispositivos disponibles.
- Análisis y elección del programario a utilizar.
- Configuración y establecimiento de comunicación entre los dispositivos.

- Creación y configuración de la red mediante un servidor de red LoRaWAN.
- Programación de aplicaciones en lenguajes como JavaScript y VBScript (Visual Basic Script).
- Traspaso de los datos obtenidos en el gestor de red a un servidor OPC.
- Desarrollo de una aplicación SCADA para la visualización de los datos.

Con la consecución de estas acciones, se quiere conseguir una red LoRaWAN con el siguiente flujo de comunicaciones:

1. Los sensores LoRa realizarán lecturas analógicas o digitales, las cuales se enviarán a una puerta de enlace.
2. La puerta de enlace reenviará estos mensajes a un servidor de red LoRaWAN.
3. El servidor de red LoRaWAN, gestionará el envío de estos mensajes a las aplicaciones.
4. Una aplicación programada en JavaScript acondicionará los mensajes para su utilización en los software destino.
5. Estos mensajes acondicionados llegarán a un servidor de datos, el cual los hará accesibles mediante el protocolo OPC a una aplicación SCADA.
6. La aplicación SCADA utilizará estos datos para monitorizar y controlar un proceso industrial simulado. Además desde esta se deberá poder hacer llegar mensajes a los nodos, consiguiendo una comunicación bidireccional en la red LoRaWAN.

1.4. Requerimientos

La puesta en marcha del prototipo de red privada LoRaWAN tiene una serie de requerimientos previos, los cuales se citan a continuación.

A nivel de conocimientos generales se requiere conocer:

- Funcionamiento y aplicación de la tecnología LoRa. Esta es una tecnología inalámbrica con un rango de alcance de 20km en campo abierto. Con una velocidad de transmisión de datos de un máximo de 50 kb por segundo, el consumo de energía de esta especificación es mínimo. Una de sus mayores ventajas es el hecho de operar en bandas de frecuencia libres (868MHz en Europa), lo que supone libertad a los usuarios para crear sus propias redes privadas LoRaWAN.

- Características y implementación de protocolos M2M (Machine to Machine). Este término genérico engloba los intercambios de información entre máquinas.

A nivel material y de recursos, se requiere disponer de los siguientes equipos LoRa y softwares industriales:

- Atim ACW/LW8-TH: sensor LoRa para interiores que realiza lecturas analógicas de temperatura y humedad en intervalos de un mínimo de 10 minutos. Este tiene una resolución de 0.01°C y una precisión de $\pm 0.3^\circ\text{C}$ para la temperatura. Para la humedad cuenta con una resolución de 0.025 %RH (humedad relativa) y una precisión de $\pm 3\%$ RH [3].
- Atim ACW/LW8-WL: sensor LoRa que realiza lecturas digitales de presencia de agua a través de sus dos terminales frontales. Estas son enviadas cada 10 minutos a las puertas de enlace dentro del rango [4].
- Atim ACW/LW8-LVL: sensor LoRa destinado a la medición del nivel de llenado de tanques, contenedores o silos. Este realiza lecturas analógicas en intervalos de como mínimo 4 minutos mediante ultrasonidos, en un rango de 20cm a 5.5m de distancia con una resolución de 1cm [5].
- Actility ODU Macro V1.5 - 8 channels: puerta de enlace LoRaWAN que cuenta con 8 canales, es decir, es capaz de comunicar con 8 dispositivos. Aguas abajo, comunica mediante la especificación LoRa con los nodos sensores. Aguas arriba se conecta a una red local mediante TCP/IP (Transmission Control Protocol/Internet Protocol) Ethernet. Su principal función es la de reenviador de paquetes. Esto consiste en reenviar los mensajes que le llegan de los sensores al servidor de red LoRaWAN y viceversa [6].
- ThingPark Enterprise: servidor de red LoRaWAN desde el cual se gestionan los mensajes de la puerta de enlace, determinando sus aplicaciones destino así como otras funciones. Este cuenta de una interfaz web que permite a los usuarios su uso desde el navegador [7].
- Node-RED: herramienta de desarrollo basada en programación visual de flujos para conectar dispositivos, APIs (Application Programming Interface) y servicios online, principalmente orientada a aplicaciones IoT. Esta permite al usuario programar funciones JavaScript mediante un editor de flujos amigable y accesible [8].
- KEPServerEX: servidor que recoge datos de los dispositivos de campo mediante una amplia variedad de controladores de comunicación (Modbus, Siemens, etc.) y los hace accesibles a los clientes finales, como podría ser una aplicación SCADA, a través del estándar de comunicación OPC [9].

- AVEVA Edge 2020: software de creación de aplicaciones SCADA para la monitorización de procesos industriales. Este cuenta con más de 250 controladores que permiten una comunicación fluida entre el HMI (Human Machine Interface) y todos los principales fabricantes de PLC (Programmable Logic Controller) y dispositivos de campo [10].

1.5. Justificación

La elección de este proyecto vino marcada en gran parte por mi situación y la de la empresa en la que me encontraba haciendo las prácticas.

Logitek es una empresa distribuidora que ofrece soluciones a sus clientes dentro del ámbito de la automatización industrial, monitorización de procesos, ciberseguridad y comunicaciones M2M entre otros. En esos momentos, estaba siendo formado especialmente en comunicaciones industriales y su implementación en las plantas.

Con el auge del concepto IIoT (Industrial Internet of Things), Logitek está analizando productos del mercado para crear soluciones completas dentro de este sector. Una de las tecnologías IoT en las que está más interesada es LoRaWAN. Es por esto, que en las oficinas se empezaron a adquirir equipos de fabricantes relevantes como Atim y Actility.

Con objetivo de llegar a esta solución completa, nuestro Solution Manager Fernando Campos me ofreció los equipos para estudiar la comunicación entre ellos y su implementación en una red privada LoRaWAN. Además, se quería valorar la integración de esta red en los sistemas comunmente encontrados en la industria convencional. Finalmente se pretendía complementar el estudio teórico y de mercado que la empresa ya estaba realizando sobre la tecnología LoRa.

Este proyecto, al tratar un concepto tan de actualidad en el sector y estar directamente relacionado con los conocimientos que estaba adquiriendo, me pareció una gran idea para seguir creciendo profesionalmente.

Por ello, el capítulo 2 desarrolla la situación actual relacionada con la tecnología LoRa y presenta los principales productos que existen en el mercado para el desarrollo de esta tecnología en el marco de la industria 4.0. El capítulo 3 justifica la plataforma escogida mientras que el capítulo 4 redacta el procedimiento seguido en la creación de la red y el capítulo 5 comenta los resultados de la plataforma desplegada. Adicionalmente, en el apartado 6 se analiza el coste de despliegue de la red. Para terminar, el apartado 7 recoge las conclusiones más relevantes del proyecto, así como una valoración personal del futuro de la tecnología LoRa.

Capítulo 2

Estado del arte

2.1. LoRaWAN y la especificación LoRa

2.1.1. Contexto y definición

Para comprender que es LoRaWAN, primeramente es conveniente definir el concepto LoRa. LoRa es una tecnología de modulación de frecuencia creada por el fabricante Semtech. Esta es un ejemplo de tecnología inalámbrica, como lo son WiFi (Wireless Fidelity) o Bluetooth, capaz de intercambiar pequeñas cantidades de información con bajo consumo y alta cobertura. De hecho, el término LoRa está conformado por el inicio de las palabras Long Range. Esta modulación utiliza la técnica de espectro ensanchado, en la cual las señales utilizan más ancho de banda del necesario teóricamente. Esto permite la recepción de múltiples señales al mismo tiempo que tienen distintas velocidades de datos. Corresponde al nivel 1 del modelo OSI (Open Systems Interconnections), es decir, el medio físico.

LoRaWAN, en cambio, es el protocolo de comunicación para redes LPWAN (Low Power Wide Area Network) que se basa en la tecnología LoRa. La principal funcionalidad de este es interconectar dispositivos cumpliendo los requisitos clave del Internet of Things, como la comunicación bidireccional, la seguridad de punto a punto, movilidad y servicios de localización [11]. [Figura 2.1]



Figura 2.1: Comunicación mediante LoRaWAN [12]

La LoRa Alliance es una asociación de compañías y empresas que unen sus capacidades para promover y contribuir al despliegue a gran escala de las redes LoRaWAN. Está es una alianza sin ánimo de lucro en la que los miembros se benefician de un ecosistema de colaboradores activos que ofrecen soluciones, productos y servicios que crean oportunidades nuevas y sostenibles [13].

2.1.2. Características

En este apartado se listan y describen brevemente las características más destacables de las redes LoRaWAN:

- **Largo alcance:** bajo condiciones óptimas el alcance de LoRaWAN puede llegar a ser de unos 20km en sectores rurales, y unos 5km en sectores urbanos. Pese a esto, como la mayoría de tecnologías inalámbricas, el alcance está directamente afectado por los obstáculos que se encuentren en el medio. Aún así es una de las opciones más viables en zonas urbanas densas o el interior de edificios [14].
- **Baja velocidad de datos:** los dispositivos LoRa transmiten datos en un rango de 0.3kb por segundo a 50 kb por segundo. Esta velocidad es variable, y es determinada por las puertas de enlace teniendo en cuenta el rango de comunicación y la duración del mensaje.
- **Sin límite en el envío de mensajes:** el número de mensajes que pueden enviarse en las redes LoRaWAN es ilimitado. Respecto a la longitud de los paquetes de datos transmitidos por LoRa, esta tiene un valor máximo de 243 bytes.
- **Bandas de frecuencia libres:** LoRa opera comunmente en las bandas de frecuencia 433MHz, 868MHz y 915MHz. El uso de cada una de estas depende de la región donde se despliega la red. En el caso de Europa la más utilizada es la banda 868MHz. Una banda de frecuencia libre es aquella en la cual cualquier persona puede transmitir datos, sin necesidad de contar con un permiso.
- **Mínimo consumo de batería:** la mayoría de los dispositivos LoRa no necesitan alimentación externa, ya que suelen contar con baterías. Dado que la transmisión de paquetes de datos se realiza unas pocas veces al día y se transmite muy poca información, la batería de estos tiene un promedio de duración de 10 años.
- **Gran capacidad:** las redes LoRaWAN pueden soportar grandes cantidades de mensajes, del rango de millones. Esta capacidad está únicamente limitada por el número de gateways desplegados en la red, y el número de canales de estos mismos [15].

- **Fácil implementación:** la independencia alámbrica de los nodos LoRa hace fácil su implementación, sin afectar a otras redes o sistemas ya instalados. Además la capacidad de los gateways para comunicar con varios nodos permite que con unas pocas puertas de enlace se pueda crear una amplia red de dispositivos con un gran rango de cobertura.
- **Geoposicionamiento:** todos los dispositivos LoRa pueden ser geolocalizados, sin costes o consumos de batería adicionales, siempre que el gateway con el que estén comunicando soporte la función. Esto es posible por medio de 2 métodos, el indicador de fuerza de la señal recibida (RSSI) y la diferencia en tiempo de llegada (TDoA), siendo este último el más preciso.
- **Seguridad:** los paquetes de datos se transmiten cifrados en todo su recorrido, es decir, desde que salen de los nodos hasta que llegan al Network Server. Este cifrado se basa en el algoritmo AES (Advanced Encryption Standard), esquema de cifrado por bloques validado como un estándar por el gobierno de los Estados Unidos [16].

También cabe destacar los siguientes parámetros que definen la modulación LoRa:

- **Ancho de banda:** es el rango de frecuencia que utiliza la señal portadora de los datos (chirp).
- **Factor de propagación (spreading factor):** este valor define cuántas señales portadoras de datos se envían por cada segundo. Este es determinado por la red según las condiciones ambientales entre el dispositivo y el gateway.
- **Tasa de codificación (coding rate):** define el método de codificar para corrección de errores. Este añade bits de control para determinar si los datos son correctos o no. Debe de modificarse según las condiciones del canal utilizado, por ejemplo, si hay muchas interferencias se recomienda aumentar este parámetro.
- **Canal:** frecuencia central de la banda de frecuencia que representa el canal a usar [11].

2.2. Estructura de las redes LoRaWAN

La estructura de una red LoRaWAN estandar, está conformada por los siguientes 4 elementos:

- **Nodos:** se les llama nodos a los dispositivos sensores o actuadores que comunican mediante la modulación LoRa con las puertas de enlace.
- **Puerta de enlace:** este dispositivo es el encargado de recopilar la información de los distintos nodos con los que tiene establecida comunicación, y enviar estos datos al correspondiente Network Server.

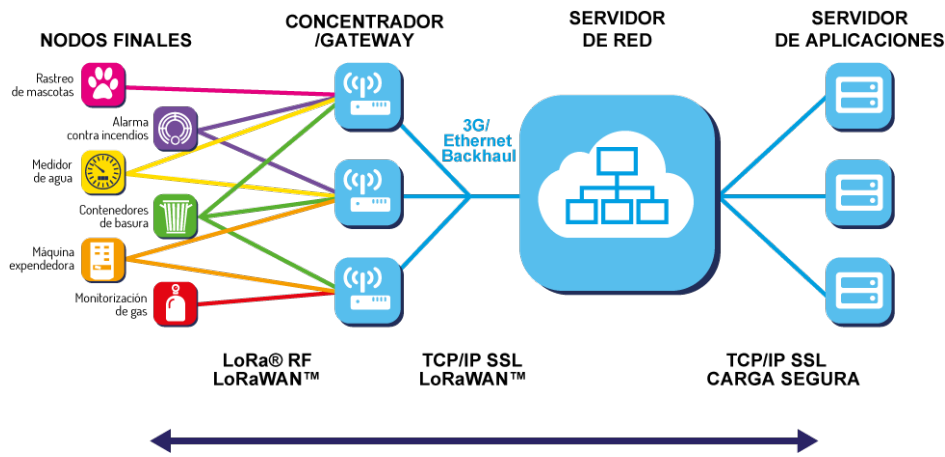


Figura 2.2: Estructura estándar de las redes LoRaWAN [18]

- Network Server: es la aplicación encargada de gestionar la red LoRaWAN en su totalidad. Este suele ser accesible vía url.
- Application Server (servidor de aplicaciones): este término engloba las diferentes aplicaciones finales encargadas de administrar, visualizar e incluso manipular los datos que llegan desde los nodos.
- Join Server: este elemento tiene como principal función gestionar el proceso de activación por aire (OTAA) para que los nodos puedan ser agregados a la red LoRaWAN [17].

En el esquema [Figura 2.2] se puede apreciar la comunicación establecida entre los diferentes elementos de la red. En las redes LoRaWAN, las tramas de datos pueden transmitirse a través de los dispositivos tanto aguas arriba (desde los nodos hasta los Application Server) como aguas abajo (desde los Application Server hasta los nodos). Los paquetes de información transportados aguas arriba reciben el nombre 'uplinks'. Estos contienen la información recogida por el nodo. En cambio, los paquetes transportados aguas abajo son denominados 'downlinks'. Estos pueden contener comandos de control de red o modificaciones sobre los parámetros de configuración del nodo [19].

Pese a que esta estructura es la más habitual, en casos de redes privadas con pocos dispositivos esta puede variar. Por ejemplo, si se cuenta con únicamente una puerta de enlace probablemente no sea necesario disponer de un Network Server para gestionar la red. Uno de los requisitos para poder obviar este elemento es que el gateway soporte los mismos protocolos de comunicación que las aplicaciones finales [20].

En los siguientes subapartados se profundiza sobre las características de cada uno de estos elementos, así como su funcionalidad.



Figura 2.3: Ejemplo de nodo LoRaWAN [21]

2.2.1. Nodos

Estos dispositivos son el elemento más bajo de una red LoRaWAN. Estos suelen ser sensores o actuadores. Debido a las limitaciones de transporte de datos de la tecnología LoRa estos suelen leer únicamente una o dos variables, como pueden ser la presencia de agua, la temperatura o la humedad. Son los nodos los que generan los paquetes de datos, con la información que nos interesa transportar hasta las aplicaciones finales, donde se le dará uso.

Estos se clasifican en 3 grupos diferentes, según su funcionalidad y comportamiento. En cambio, teniendo en cuenta su método de activación, se pueden clasificar en 2 grupos. En los próximos apartados se entra en detalle en cada una de estas clasificaciones.

2.2.1.1. Clases

Los nodos compatibles con la tecnología LoRa se clasifican en las siguientes 3 clases:

2.2.1.1.1. Clase A Esta es la clase predeterminada de los nodos LoRa, lo que significa que todo dispositivo final con tecnología LoRa ha de soportar esta especificación.

En la clase A el nodo siempre es quién inicia la comunicación, y esta es totalmente asíncrona. El proceso de comunicación comienza con el envío de los datos por parte del nodo. A continuación se dejan dos breves ventanas para la recepción de downlinks. Una vez finalizadas estas ventanas, el dispositivo entra en modo suspensión hasta el envío programado del siguiente uplink.

Este breve intervalo de para recibir los downlinks puede parecer un inconveniente. Sin embargo, es aquí donde el Network Server muestra una de sus funcionalidades. Cuando una aplicación final envía un downlink aguas abajo a un dispositivo clase A, el Network Server se encarga de almacenar temporalmente estos datos, y no enviarlos hasta que no recibe el uplink programado del nodo. Con esto se asegura que los downlinks lleguen al nodo dentro de estas ventanas de recepción.

Debido a este funcionamiento, la clase A es el modelo de menor consumo, al mismo tiempo que permite el envío de uplinks en cualquier momento [22].

2.2.1.1.2. Clase B Los nodos de clase B tienen el mismo comportamiento que los de clase A, con la adición de estar sincronizados con la red mediante balizas periódicas. Además, este tipo de dispositivos tienen la posibilidad de establecer ventanas programadas para la recepción de downlinks, sin que estas estén sujetas al envío de uplinks. Estas ventanas adicionales se pueden programar con intervalos de hasta 128 segundos, según las necesidades del cliente.

Esta capacidad supone un consumo extra de la batería del nodo. Aún así, esta diferencia sigue siendo lo suficientemente baja para ser asumida por dispositivos alimentados por baterías [23].

2.2.1.1.3. Clase C Los nodos comprendidos en la clase C son aquellos que tienen un modelo de envío de uplinks como los nodos tipo A y B, pero su receptor está permanentemente activo para la recepción de downlinks. Es decir, el Network Server puede enviar mensajes descendentes a estos dispositivos en cualquier momento.

Obviamente este comportamiento implica un consumo mucho mayor de energía, de aproximadamente 50mW. Por ello, este suele ser adoptado por dispositivos alimentados externamente.

Algunos dispositivos alimentados con baterías son compatibles con los comportamientos de las clases A y C. En estos casos, el modelo de gran consumo suele estar reservado para realizar actualizaciones de firmware a través del aire.

2.2.1.2. Activación

Para la hacer posible la dada de alta de los nodos en una red LoRaWAN, estos suelen venir con una serie de claves pre-programadas en su software. Antes de pasar a los métodos de activación de estos, es importante definir y ponerle nombre a algunas de estas claves:

- Device Extended Unique Identifier (DevEUI): código de identificación del dispositivo final. Consta de 64 bits.
- Device Address (DevAddr): dirección del dispositivo final. Consta de 32 bits.
- Application Extended Unique Identifier (AppEUI): código de identificación de la aplicación final. Consta de 64 bits.
- Application Key (AppKey): clave usada en el método de activación por aire para generar las claves de sesión.
- Network Session Key (NwkSKey): encripta los metadatos incluidos en los paquetes de datos LoRa.

- Application Session Key (AppSKey): encripta los datos relevantes (payload) del paquete de datos LoRa [24].

2.2.1.2.1. Activación por aire (OTAA) Los nodos que únicamente la activación por aire vienen con tres códigos internamente pre-programados. Estos son el Device EUI, el Application EUI y el Application Key, los cuales el fabricante ha de otorgar al cliente con la compra de cada dispositivo final.

El proceso de activación por aire sigue los siguientes puntos:

1. El nodo transmite la solicitud de unión junto con su DevEUI, AppEUI y AppKey. A esto se le añade por motivos de seguridad un DevNonce, el cual es un número aleatorio que se utiliza en mensajes encriptados para evitar que la solicitud de unión pueda ser recreada por otros dispositivos [25].
2. Cualquier puerta de enlace que reciba este paquete de datos lo reenvía a su Network Server.
3. El Network Server recibe la solicitud y consulta la entidad asociada con el AppEUI para validarla. Si la solicitud es permitida, este responde enviando un mensaje de aceptación de unión (join-accept). Este mensaje contiene un identificador de red (NetID), un DevAddr y un AppNonce, el cual permite al nodo generar las claves de sesión.
4. El gateway con la mayor calidad de señal con el nodo envía el mensaje.
5. El nodo almacena el NetID, el DevAddr, la configuración de la red, y usa el AppNonce para generar las claves de sesión NwkSKey y AppSKey.

Una de las ventajas de este método es que en el caso que se quiera cambiar el nodo a una nueva red LoRaWAN, este no necesita ser reprogramado, sino que genera unas nuevas claves de sesión.

2.2.1.2.2. Activación por personalización (ABP) La activación por personalización es un método en el que el nodo no necesita las claves DevEUI, AppEUI y AppKey. A diferencia de los dispositivos OTAA, estos vienen pre-programados con las claves de sesión NwkSKey y AppSKey.

Este método no requiere de un proceso para dar de alta el nodo en la red LoRaWAN. Simplemente se ha de configurar previamente en el Network Server las claves DevAddr, AppSKey y NwkSKey, para que este pueda reconocer las transmisiones del dispositivo final [26].

El principal inconveniente de este método es su seguridad. Si en cualquier momento las claves de sesión son descubiertas, incluso antes de la activación, estas pueden ser utilizadas con otro dispositivo. Tanto este caso, como el pasar a una nueva red LoRaWAN o la expiración de las claves, implica el tener que reprogramar el nodo.



Figura 2.4: Ejemplo de puerta de enlace LoRaWAN [28]

2.2.2. Gateway

La puerta de enlace de una red LoRaWAN es el dispositivo encargado de recibir los mensajes radiofrecuencia de los diferentes nodos dentro de su rango de cobertura y enviarlos al servidor de red.

La función principal de un gateway es la de ‘reenviador de paquetes’ (packet forwarder). Packet forwarder es un programa que corre en el procesador de la puerta de enlace cuya única función es reenviar los paquetes recibidos. Para comunicar con los nodos, este interactúa con el chip LoRa dentro del gateway para recibir y emitir paquetes de datos LoRa. Cuando recibe o transmite datos del servidor de red, este suele utilizar medios físicos como un cable Ethernet, Wi-Fi y 3G o 4G entre otros, dependiendo de las características del gateway. En el envío de estos uplinks al Network server, también se añaden parámetros con datos referentes a la recepción del mensaje, como el RSSI (indicador de fuerza de la señal recibida). Para los mensajes descendentes de LoRaWAN, una puerta de enlace ejecuta las solicitudes de transmisión que provienen del servidor de red sin ninguna interpretación de la carga útil [27].

Otra de las funcionalidades de este dispositivo es la verificación de la integridad de los paquetes de datos. Para la información proveniente de los nodos, se analiza el CRC de cada mensaje (Cyclic Redundancy Check). Cuando este algoritmo de corrección de errores indica una correcta integridad del mensaje, este es enviado al Network Server. En el caso contrario el mensaje es descartado [29].

Los gateways no tienen ninguna clase de vínculo fijo con los nodos. Los mensajes emitidos por los nodos viajan a través de la frecuencia establecida y, en caso de que un gateway esté dentro del rango de alcance, este recibirá el mensaje. En casos de redes LoRaWAN con varias puertas de enlace un mismo uplink proveniente de un nodo puede ser recibido por varias puertas de enlace. Esto permite reducir drásticamente la tasa de errores y pérdida de paquetes, ya que es complicado que los mensajes emitidos por un dispositivo final no lleguen a al menos un gateway.

En el mercado, los gateway estándar suelen contar con 8 o 16 canales LoRa. Cada uno de estos canales, permite la comunicación con un único dispositivo final.

2.2.3. Servidor red

El LoRaWAN Network Server es el elemento que administra la red en su totalidad. Entre sus funciones más destacadas se encuentran:

- La comprobación de las direcciones de los dispositivos.
- El ajuste de la velocidad de datos (data rate).
- El reconocimiento de los mensajes recibidos.
- El reenvío de los uplinks al Application Server correspondiente.
- El almacenamiento de los downlinks en la cola, hasta que se abre la ventana de recepción del dispositivo final destinatario.
- El reenvío de mensajes relevantes para la validación de los dispositivos en la red LoRaWAN [30].

Una de las funciones del servidor de red es asegurar que no se reciben mensajes duplicados de los nodos. Ya que varios gateways pueden recibir el mismo paquete de datos LoRa desde un nodo, el Network Server se encarga de quedarse con uno de los mensajes y eliminar el resto de copias.

Respecto a los downlinks, en redes con varias puertas de enlace es necesario determinar cuál de ellas realizará el envío al dispositivo final. Esto lo consigue el servidor de red mediante la comparación del parámetro RSSI (indicador de fuerza de la señal recibida), de los mensajes ascendentes provenientes del nodo. A la hora de enviar los downlinks a los nodos, el servidor de red determina como encargado al gateway que mejor señal tiene con el nodo destinatario del mensaje [31].

2.2.4. Servidor de aplicaciones

El servidor de aplicaciones gestiona la capa de aplicación LoRaWAN, incluyendo entre sus funciones el descifrado y decodificación de datos de uplinks, la cola de downlinks y el cifrado y codificación de downlinks.

En términos más simples, es el elemento encargado de administrar toda la información de los paquetes de datos y hacerla servible a las aplicaciones finales.

Este suele funcionar como servidor utilizando protocolos estándar del ámbito del Internet de las Cosas, como MQTT (Message Queue Telemetry Transport) y HTTP (Hypertext Transfer Protocol) [32].

2.2.5. Servidor de unión

El Join Server, o servidor de unión, es el elemento encargado de gestionar toda la información referente al proceso de dar de alta un dispositivo final en la red. Puede considerarse una extensión del servidor de red, ya que siempre trabaja conjuntamente con este y no es un elemento tangible del esquema.

Este indica al Network Server qué servidor de aplicaciones tiene que conectarse al nodo y realiza el cifrado de las claves de sesión de red y de aplicación. Comunica la clave de sesión de red del nodo al Network Server, así como también envía la clave de sesión de la aplicación al Application Server correspondiente [33].

2.3. Aplicaciones de LoRaWAN

2.3.1. Contexto Internet of Things

Para analizar que lugar ocupa la tecnología LoRa y las redes LoRaWAN en el mercado actual, es importante tener en cuenta el concepto del Internet de las Cosas (IoT).

En términos generales, el concepto Internet of Things engloba toda la red de dispositivos que son capaces de recopilar todo tipo de datos, y intercambiarlos con otros dispositivos y sistemas mediante Internet. La filosofía de este es tener la máxima cantidad de información para poder utilizarla con fines como la monitorización, el control, prevención de incidencias, etcétera. Como uno de los principales objetivos de esta red, se destaca la interconexión de dispositivos que pueden operar de manera semi-independiente o con una intervención humana mínima.

Este concepto está muy a la alza. Permite una gran cantidad de nuevas oportunidades de acceso a datos, lo que abre la posibilidad a desarrollar avances importantes en sectores como la educación, transporte, seguridad, agricultura y industria, entre otros.

Los dispositivos IoT pueden ir desde objetos domésticos o cotidianos hasta sensores industriales de alta complejidad [34].

2.3.2. Industrial Internet of Things

Por otro lado, el término Industrial Internet of Things (IIoT) es una sub-categoría del Internet de las Cosas, que se utiliza para referirse a toda esa tecnología IoT dentro del contexto industrial.

Es creído por mucha gente, que el establecimiento del concepto IIoT en la industria supondrá el inicio de la ‘Cuarta Revolución Industrial’, también conocida como Industria 4.0.

Teniendo claro que los beneficios que aporta la adopción del IIoT dependen de la implementación que

se haga en cada caso, a continuación se citan algunos de los beneficios generales que esto puede suponer:

- Aumento de la cantidad de datos relevantes para la monitorización y el control. Así como una mejora de la accesibilidad a estos mismos.
- Complejidad de red reducida, lo que supone una reducción de los costes de mantenimiento y una mejora en la seguridad de esta.
- Facilidades para el tratamiento de todos los datos provenientes de diferentes dispositivos desde plataformas de terceros.
- Mejora de la capacidad de automatización y la eficiencia en plantas industriales, reduciendo la necesidad de intervención humana.
- Capacidad mejorada para la prevención de fallos y incidencias, así como generación de más oportunidades para el planteamiento de operaciones [35].

Teniendo en cuenta lo descrito en los anteriores apartados, se puede deducir el por qué la tecnología LoRa y las redes LoRaWAN encajan en el mercado IoT e IIoT.

En resumidas cuentas, LoRaWAN permite la interconexión de todos estos dispositivos inteligentes, sin requerir de instalaciones de red complejas. También ofrece facilidad y libertad de desarrollo al usuario, además de la capacidad de integrar todo este flujo de información en softwares de terceros.

2.3.3. Comparación frente a otras tecnologías inalámbricas

El conjunto de tecnologías de redes inalámbricas de la actualidad puede diferenciarse mediante la comparación de 3 aspectos generales:

- El consumo energético necesario para la transmisión de los datos.
- El rango de alcance, o distancia entre nodo y antena.
- La capacidad de transmisión de datos, o data rate.

Actualmente, ninguna de las tecnologías disponible cumple de manera satisfactoria los 3 aspectos a la vez. Esto es debido al aumento proporcional de las necesidades. Por ejemplo, para transmitir paquetes de datos a mayor distancia se requiere de más consumo energético. Es por ello que las tecnologías de transmisión inalámbricas pueden agruparse según el cumplimiento de estos 3 parámetros, tal y como se aprecia en el gráfico [36].

LoRaWAN, como ya se ha mencionado anteriormente, corresponde a la categoría de redes LPWAN (Low Power Wide Area Network). Las redes LPWAN son aquellas que permiten transmitir un pequeño volumen de datos entre dispositivos y puertas de enlace separados por largas distancias, requiriendo un muy bajo consumo energético.

En los siguientes apartados se compara LoRaWAN con algunas de las tecnologías de redes LPWAN más relevantes.

2.3.3.1. Sigfox

Sigfox es un operador de red LPWAN que ofrece una solución de conectividad IoT usando sus tecnologías patentadas. Esta compañía tiene desplegadas multitud de estaciones base (gateways), que están conectadas a los servidores back-end mediante una red basada en IP. Además de la red y su mantenimiento, Sigfox deja disponibles en su Application Server los datos enviados por los dispositivos finales.

Al igual que LoRaWAN, Sigfox opera en bandas de frecuencia libres según la región, siendo estas 915MHz en América del Norte, 868MHz en Europa y 433MHz en Asia.

Los dispositivos con tecnología Sigfox utilizan modulación de banda ultra-estrecha, de 100Hz. Esta modulación permite una utilización del ancho de banda muy eficiente, reduciendo drásticamente los niveles de ruido y por ende, el consumo de energía [37].

En una red Sigfox, la cantidad de mensajes ascendentes (uplinks) que se pueden enviar es de 140 al día, o lo que es lo mismo, 1 cada 10 minutos. La longitud máxima de cada uno de estos uplinks es de 12 bytes. Respecto a los mensajes descendentes, su máximo es de 4 al día, con una longitud máxima de 8 bytes cada uno. El límite de velocidad de datos es de 100 bits por segundo. El hecho de tener estas limitaciones en cuanto al volumen de datos, hace que esta tecnología tenga un alcance de 10 km en sectores urbanos y 40 km en espacios abiertos.

2.3.3.2. NB-IoT

NB-IoT, siglas de los términos Narrow Band Internet of Things, es una tecnología de redes LPWAN para aplicaciones IoT ofrecida por grandes compañías de telecomunicaciones establecidas en el mercado de telefonía móvil, como Vodafone. A diferencia de Sigfox, la tecnología NB-IoT no está muy desplegada actualmente. La implementación de estas redes se va aumentando según los proyectos demandados. Es decir, si se tiene un proyecto interesante en cuanto a volumen de equipos, la operadora en cuestión monta una red NB-IoT dedicada a este.

Esta tecnología opera en las bandas de frecuencia licenciadas, como 700MHz, 800MHz y 900MHz. Estas bandas son las mismas utilizadas por protocolos como el sistema global para comunicaciones móviles

(GSM) y la evolución a largo plazo (LTE). NB-IoT puede coexistir con estos protocolos en estas bandas, ya que a fin de cuentas esta tecnología se basa en el protocolo LTE. Tal es así, que la tecnología NB-IoT puede integrarse con la infraestructura LTE existente, simplemente realizando una actualización de su software. Esta permite la comunicación de sus dispositivos en rangos de alcance de 1 km en sectores urbanos y 10 km en espacios abiertos [38].

Los dispositivos con tecnología NB-IoT, como su nombre indica, utilizan modulación de banda estrecha, concretamente de 200KHz. Esta corresponde a un bloque de recursos en la transmisión GSM y LTE.

En las redes NB-IoT la cantidad de mensajes no está limitada. En cuanto a la longitud máxima de estos, el límite es de 1600 bytes, tanto para los mensajes ascendentes como para los mensajes descendentes. El límite de velocidad de datos es de 200 kilobits por segundo.

2.3.3.3. Ventajas y desventajas

Habiendo visto las características generales y técnicas de las redes LPWAN más importantes, en este apartado se destacan las ventajas y desventajas de LoRaWAN respecto al resto.

A nivel de usuario, el hecho de que LoRaWAN tenga la posibilidad de no estar ligado a operadoras, es probablemente la característica más diferencial. Esto supone una gran libertad a la hora de desarrollar y implementar una red a gusto del consumidor. De hecho, LoRaWAN es el único que permite la implementación de redes privadas, en las cuales el usuario es el responsable de autogestionarlas. Además de libertad, esto reduce el coste de la red. El usuario se ahorra pagar los servicios de las operadoras y el coste de añadir nuevos dispositivos a su red, lo que resulta en una de la escalabilidad. También es importante destacar que las redes LoRaWAN y NB-IoT cuentan con encriptado en la transmisión de los paquetes de datos, mientras que Sigfox no.

En cuanto al envío de datos, LoRaWAN se posiciona junto a NB-IoT como la mejor opción para aplicaciones que necesiten un gran número de envío de mensajes al día, debido a que este no está limitado. NB-IoT ofrece una mayor longitud máxima en sus paquetes de datos, aunque los 243 bytes de los mensajes LoRaWAN es más que suficiente para la mayoría de usos en el ámbito IoT [39].

Si comparamos el rango de alcance de estos tipos de redes LPWAN, LoRaWAN es la opción intermedia. Sin embargo su fácil y más barata escalabilidad hace más viable la integración de nuevos dispositivos para aumentar su alcance. En redes con un bajo volumen de mensajes y una gran distancia entre dispositivos, la elección de Sigfox sería la que necesitaría menos nodos y puertas de enlace.

Por último, en los tres casos los nodos permanecen en estado inactivo la mayor parte del tiempo. Sin embargo, los nodos NB-IoT se comunican de manera síncrona, lo que supone un mayor consumo de energía y una reducción de la duración de la batería respecto a los dispositivos finales LoRaWAN y Sigfox.

Cuadro 2.1: Tabla comparativa de las tecnologías de redes LPWAN

	LoRaWAN	Sigfox	NB-IoT
Permite red privada	Sí	No	No
Rango de alcance	5 km (urbano), 20 km (rural)	10 km (urbano), 40 km (rural)	1 km (urbano), 10 km (rural)
Máx. mensajes/día	Ilimitado	140 uplinks, 4 downlinks	Ilimitado
Velocidad de datos máx.	50 kb/s	100 b/s	200 kb/s
Velocidad de datos adaptativa	Sí	No	No
Máx. longitud del payload	243 bytes	12 bytes (uplink), 8 bytes (downlink)	1600 bytes
Frecuencias operativas	868 MHz en Europa, 915 MHz en Norte América, 433 MHz en Asia	868 MHz en Europa, 915 MHz en Norte América, 433 MHz en Asia	Bandas LTE licenciadas
Ancho de banda	125 kHz / 250 kHz	100 Hz	200 kHz
Autenticación y encriptado	Sí (AES 128b)	No	Sí (encriptado LTE)
Inmunidad a interferencias	Muy Alta	Muy alta	Baja
Geolocalización	Sí (RSSI y TDoA)	Sí (RSSI)	No

En el cuadro 2.1, se comparan los aspectos más relevantes de las tres tecnologías de redes LPWAN vistas anteriormente.

2.3.4. Ejemplos de uso

El uso actual de LoRaWAN está bastante ligado a los sectores que están adoptando la filosofía IoT. Algunos de los ámbitos en los que más se están implementando redes LoRaWAN son:

- Agricultura inteligente
- Ciudades inteligentes



Figura 2.5: Usos comunes de LoRaWAN [41]

- Control industrial inteligente
- Automatización industrial
- Medidores inteligentes
- Cadenas de suministro y logística inteligente
- Sanidad inteligente
- Medio ambiente inteligente [40]

La implementación de tecnologías IoT en la industria está ligada a la monitorización de todos aquellos datos que no son críticos ni requieren de procesamiento en tiempo real. Esta gran cantidad de variables toman relevancia si son contextualizadas, y el acceso a ellas nos aporta una serie de ventajas y oportunidades de mejora. Especialmente, el ecosistema IIoT cobra más sentido en lugares de la industria donde no es viable llegar con sensórica cableada. Su mayor alcance frente a tecnologías como el WiFi, su fácil despliegue, y su bajo consumo y coste, hacen de esta la opción óptima para cubrir necesidades específicas.

A nivel nacional, las tecnologías IoT aplicadas a la Industria está cogiendo fuerza, con muchas empresas ayudando en la implementación de proyectos con redes LPWAN. El mejor ejemplo de esto es la compañía Cellnex, que desde hace años actúa como operadora de la tecnología Sigfox en España, cubriendo la mayor parte del territorio. Un par de ejemplos de sus aplicaciones generales en la industria han sido:

- Eficiencia energética, ayudando a aumentar el número de contadores de consumos energéticos en la red. La contextualización de esta información ha aumentado la capacidad de optimización de cada equipo.

- Mantenimiento, permitiendo la monitorización online de los activos se ha ayudado a mejorarlo y optimizarlo. Se ha obtenido mejora en los tiempos de respuesta, en la prevención de errores y el mantenimiento basado en condiciones y predictivo.

Si bien es cierto que LoRaWAN está adquiriendo un papel importante en las industrias internacionales, en España la aplicación de la tecnología en el sector industrial está en una fase muy alfa. Esto es debido tanto por falta de popularidad como por desconocimiento.

Aún así, ya hay industrias que han comenzado a implementar proyectos de redes LoRaWAN. Un ejemplo a destacar es la compañía Redexia, la cual actúa como operadora ofreciendo cobertura y soluciones LoRaWAN según las necesidades de los clientes. Esta cuenta con la mayor red LoRaWAN a nivel nacional [42].

2.3.4.1. Proyecto de monitorización del flujo de aire en almacenes refrigerados de fruta

Este fue un proyecto de investigación realizado por personal del Instituto de Microsensores, en la Universidad de Bremen. La aplicación fue implementada en un almacén de fruta, en este caso manzanas, con capacidad para almacenar hasta 40 toneladas de el producto. Este volumen está repartido en multitud de contenedores apilados uno encima del otro, en una estructura de filas y columnas.

Un requisito fundamental en este tipo de almacenes es el mantenimiento de temperatura y velocidad del aire frío entre las grandes cantidades de producto. Por ello, uno de los objetivos de la implementación de la tecnología LoRa en este caso fue la monitorización de estos parámetros. Con esto se quería prevenir el deterioro de las manzanas, gracias a la posibilidad de detectar puntos específicos calientes o de estancamiento de la velocidad de aire frío.

Sin embargo, el objetivo principal de este estudio fue la reducción del consumo de energía. Aproximadamente un 40% del consumo de energía de este almacén es dedicado al funcionamiento de los ventiladores del sistema de refrigeración. Los datos obtenidos mediante la tecnología LoRa permitirían conocer el uso exacto que se debería dar a estos ventiladores.

Uno de los principales inconvenientes para el uso de sensoría inalámbrica encontrados en estos almacenes es que el producto tiene altos porcentajes de agua en su cuerpo, lo que dificulta mucho las comunicaciones. De hecho, antes de implementar la tecnología LoRa en el almacén se habían probado otros dispositivos radiofrecuencia, como por ejemplo los nodos TelosB de 2.4 GHz. Estos dieron resultados muy pobres ya que, debido a su corto alcance y su topología de red de malla, se requería colocar un nodo cada 0.5 metros entre los contenedores. Aún así esto se tradujo en un gran índice de pérdida de paquetes.

Los transmisores LoRa fueron acoplados a pequeñas placas conectadas a sensores, concretamente ane-

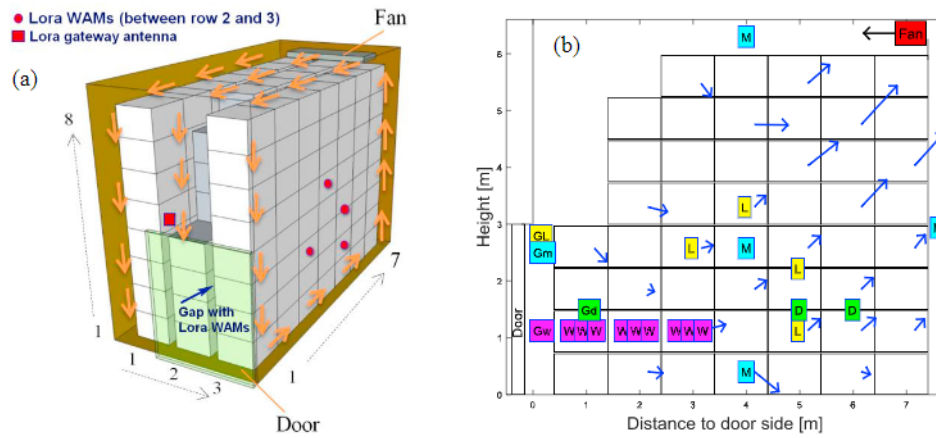


Figura 2.6: Disposición de los transmisores LoRa [43]

mómetros para medir la velocidad y dirección del flujo de aire y sensores de temperatura. Estos se colocaron en la disposición que se puede observar en la imagen [Figura 2.6], en los huecos entre las columnas de contenedores, con un único Gateway colocado en la puerta del almacén de 3x7x8m de dimensión.

Los resultados fueron muy satisfactorios. Los 2 transmisores LoRa más cercanos al gateway, a 3 y 4 metros de distancia, transmitieron todos los paquetes sin ninguna pérdida. En cuanto al más alejado, a unos 5m aproximadamente, tuvo un promedio de 2 de cada 1000 paquetes perdidos. Esto demostró la viabilidad de un solo gateway LoRa para dar cobertura a todos los transmisores de este almacén de manzanas [43].

2.3.4.2. Recopilación de datos de activos remotos

Este caso de aplicación fue implementado por la compañía Advantech para uno de sus clientes. Advantech es una empresa que ofrece soluciones IoT, especialmente aplicadas a la Industria 4.0, con algunos objetivos como el aumento de eficiencia en las plantas.

Este cliente de Advantech necesitaba recoger datos desde activos remotos. Estos activos, ya contaban con sensores inteligentes conectados capaces de servir datos sobre el consumo de gas, agua y electricidad, los cuales comunicaba mediante una interfaz Modbus. El principal problema en este caso particular era la distancia hasta la sala de control, así como la cantidad de paredes, edificios y obstáculos que hay entre ambos puntos. El coste que suponía la instalación de cableado hasta estos activos remotos era inviable.

Antes de implementar la solución LoRa, la empresa había estado recopilando estos datos de manera manual. Esto representaba un elevado coste laboral y la posibilidad de errores de lectura en el sistema.

El inconveniente fue resuelto instalando un nodo Wizzard LoRaWAN. Este nodo permite leer señales analógicas, digitales o datos de sensores Modbus RTU, y transmitirlos a una puerta de enlace LoRaWAN. La puerta de enlace recoge los datos recibidos mediante LoRa y los entrega al sistema de control SCADA en formato Modbus.

Este ejemplo es una muestra clara de las ventajas del uso de LoRa en entornos industriales en los que se cuenta con grandes cantidades de información, pero su lectura no es simple o económicamente viable [44].

2.3.4.3. Lectura de contadores

Este caso de aplicación fue conjuntamente desarrollado por las compañías Logitek y Cefalux. Logitek es una distribuidora de hardware y software con aplicaciones como control industrial, comunicaciones M2M y automatización, la cual ofrece soluciones LoRa para la implementación de redes privadas LoRaWAN. Cefalux es una integradora de proyectos relacionados principalmente con la automatización industrial.

El proyecto fue desarrollado para la empresa Mercedes, concretamente para su departamento de Servicios Generales, el cual se encarga de atender a las necesidades conjuntas de sus diferentes plantas. Este cliente dispone de centenares de contadores en sus instalaciones.

Este cliente requería de monitorizar de manera conjunta las lecturas de todos estos contadores, ya que hasta ahora este proceso se realizaba de manera manual. La idea era disponer de todos estos datos en sus softwares habituales de control, para poder actuar en base a estos con fines de optimización energética. Debido a que se quería que la monitorización de estos datos fuese constante, se necesitaban dispositivos externamente alimentados, capaces de transmitir datos de manera continua.

Como solución se implementaron los siguientes dispositivos:

- ACW-DINDA: nodo LoRa del fabricante Atim con la capacidad realizar lecturas de dos entradas digitales y una analógica. También dispone de una salida capaz de alimentar un sensor.
- ACW-DINRSM: nodo LoRa del fabricante Atim con la capacidad de realizar lecturas y escrituras en hasta 49 registros modbus.
- ACW-DIND88: nodo LoRa del fabricante Atim con 8 entradas digitales y 8 salidas digitales. [Figura 2.7]
- Industrial LoRaWAN Gateway: gateway de Atim para el interior de plantas industriales. Este cuenta con la capacidad de comunicar aguas abajo mediante LoRa y aguas arriba mediante WiFi, TCP/IP y MQTT.

Como este era un proyecto piloto para valorar la implementación de dispositivos LoRa y únicamente se utilizó un gateway, los datos no eran enviados a ningún Network Server. Gracias a la capacidad de la puerta de enlace para transmitir datos por protocolo MQTT, estos eran enviados directamente a las aplicaciones finales.



Figura 2.7: ACW-DIND88 [45]

2.3.4.4. Rastreo y localización de activos

El siguiente proyecto de implementación de LoRaWAN fue aplicado por la compañía Actility. Actility es la empresa líder en soluciones de conectividad IoT para todo tipo de aplicaciones. En este caso el cliente fue Volvo, fabricante de automóviles que posee multitud de fábricas alrededor del mundo. Este caso particular fue probado en una de sus fábricas de camiones en Francia, aunque tras su éxito se decidió implementar en todas las instalaciones de la compañía.

Esta fábrica fabrica un total de 72 camiones al día. Dentro de sus instalaciones hay entre 5000 y 6000 camiones ensamblados. Desde que estos son fabricados hasta que son entregados a los clientes, estos se trasladan numerosas veces a diferentes puntos de la fábrica para pasar por pruebas y exámenes de calidad. Durante estos traslados se detectaba que los empleados malgastaban mucho tiempo localizando cada camión, lo cual les impulsó a buscar una solución.

Actility fue la encargada de aportar esta solución, la cual consistió en la implementación de dispositivos basados en LoRa. Tras ser fabricado, en cada camión se insertaba este nodo, el cual gracias a las capacidades de las redes LoRaWAN podía ser rastreado y localizado en todo momento. Concretamente, se utilizó el modelo Abeeway Industrial Tracker, el cual combina las tecnologías GPS (Global Positioning System), GPS de baja potencia, capturador WiFi y la diferencia de tiempo de llegada (TDoA) de los gateway LoRaWAN para obtener una precisión exacta.

La información de los localizadores es enviada a los gateway mediante LoRa, y estos la transmiten al Network Server de Actility (Thingpark Enterprise) mediante conexión IP. Finalmente, la información de estos nodos es gestionada desde la aplicación industrial Thingworx, del fabricante Parametric Technology Corporation (PTC) [46].

2.4. Protocolos de comunicación

Un protocolo de comunicación es el conjunto de reglas y normas que permiten que dos o más dispositivos de una red se comuniquen entre ellos para transmitir y recibir datos.

En los siguientes apartados, se detallan los aspectos más significativos de dos protocolos de comunicación utilizados y relevantes en el proyecto, así como el motivo de su elección.

2.4.1. MQTT

MQTT, de las siglas Message Queue Server, es un protocolo de comunicaciones altamente basado en el protocolo TCP/IP. Sin embargo, cualquier protocolo de red que proporcione conexiones bidireccionales ordenadas y sin pérdidas puede soportar MQTT. Este fue creado en 1999 por Andy Stanford-Clark con el objetivo de tener un protocolo eficiente en el ancho de banda, ligero y con un bajo consumo de energía.

MQTT se ha convertido en uno de los protocolos estándar para redes de dispositivos IoT. Esto se debe en gran parte a su sencillez, velocidad y ligereza, aspectos clave dadas las limitaciones de los nodos IoT. La mayoría de servidores de red cuentan con la capacidad de transmitir datos vía MQTT. Incluso algunas puertas de enlace incorporan esta función, como se ha visto en apartados anteriores.

Además de esto, debido al incremento de popularidad de la Industria 4.0, muchos de los principales softwares de control y automatización industrial están incorporando la funcionalidad de comunicar mediante este protocolo.

Teniendo en cuenta las motivaciones de este proyecto, que entre otras cosas pretende demostrar la fácil integración de las redes IoT en el sector industrial, este protocolo es una de las mejores opciones [47].

2.4.1.1. Arquitectura

MQTT es un protocolo con arquitectura publicador-suscriptor (publisher-subscriber), en la cual ambos clientes se conectan a un servidor central denominado broker.

Para filtrar los mensajes que envía y recibe cada cliente, estos son organizados en topics. Un topic es una cadena de caracteres que el broker utiliza para diferenciar los paquetes de datos que corresponden a cada cliente.

En una comunicación vía MQTT los clientes primeramente se conectan al broker. Una vez establecida la conexión, un cliente publica un mensaje, conteniendo este un topic y un payload (información relevante del paquete de datos). Este mensaje, permanece en el broker. Es entonces cuando otro cliente se conecta al broker y se suscribe al topic. Este nuevo cliente recibe los datos correspondientes a ese topic. [Figura 2.8]

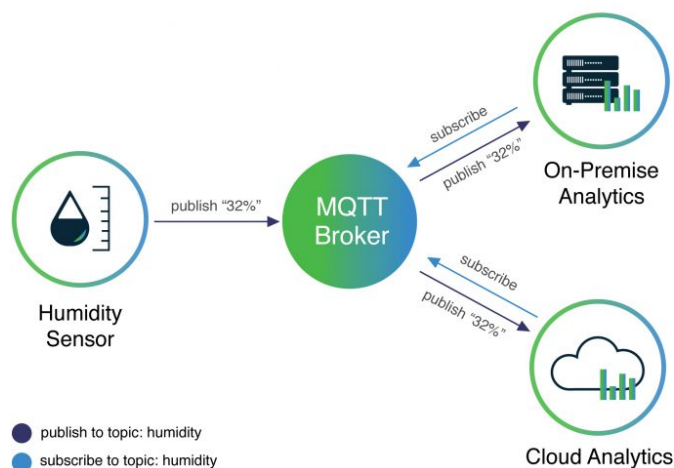


Figura 2.8: Ejemplo de arquitectura MQTT [48]

Una peculiaridad a destacar de esta arquitectura es que cualquier cliente puede hacer de publicador, suscriptor, o ambos simultáneamente. El puerto por defecto para la comunicación MQTT es el 1883 [48].

2.4.1.2. Seguridad

El protocolo MQTT presenta algunos mecanismos de seguridad en varios niveles basados en estándares. Estos son:

- Autenticación por usuario/contraseña: en un broker MQTT se permite establecer que los clientes requieran de usuario y contraseña para conectarse, así como para suscribirse a un topic. Cada cliente se identifica por un identificador único, denominado ClientId.
- Autenticación mediante intercambio de certificados x509: los certificados son documentos digitales que asocian pares de claves criptográficas con identidades, en este caso el broker y los clientes. x509 es un formato estándar para certificados de clave pública.
- Seguridad SSL/TLS: SSL y TLS, de las siglas Secure Socket Layer y Transport Layer Security, son protocolos de encriptado cuya función es cifrar el tráfico de datos entre el cliente y el servidor [49].

2.4.2. OPC UA

OPC es un estándar de comunicación en el ámbito del control y automatización de procesos basado en tecnología Microsoft. Este surgió a mediados de los años 90 como una tecnología para comunicar software HMI con los diferentes dispositivos industriales. La evolución de esta primera versión del protocolo paso a llamarse OPC DA (data access), el cual establecía como debía establecerse el intercambio de datos entre

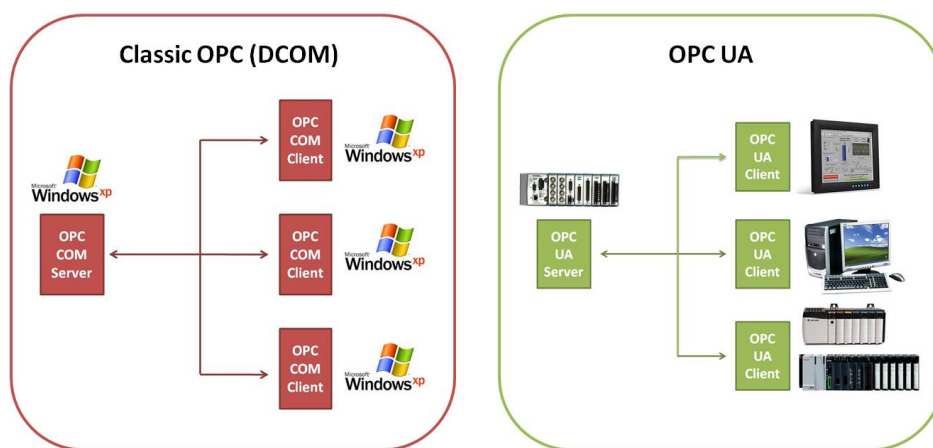


Figura 2.9: Diferencia de plataformas entre OPC DA y OPC UA [51]

clientes y servidores. Además de OPC DA, con el paso de los años fueron apareciendo diferentes versiones del protocolo según la aplicación a la que estaban destinadas. Algunos ejemplos de estas son OPC AE (Alarms and Events), para gestionar el tráfico de alarmas y eventos, y OPC XML-DA, para servicios web a través de XML (Extensible Markup Language).

OPC UA (Unified Architecture) es la evolución del protocolo estándar, que fue lanzado en 2008 para acabar con ciertas limitaciones que tenía OPC DA, incluyendo además el resto de especificaciones OPC. Este es independiente de la plataforma, el sistema operativo, el hardware y el software. Esto permite unificar la comunicación de los dispositivos industriales de diferentes fabricantes, implicando que todos los datos del proceso de producción se transfieran mediante un solo protocolo.

OPC UA es el estándar de comunicación industrial actual. Muchas industrias con procesos establecidos hace años están tratando de emigrar sus comunicaciones a este protocolo por cuestiones de comodidad y fácil escalabilidad.

Este trabajo pretende integrar las redes LoRaWAN en procesos industriales estandar en la industria. Por ello, la elección de protocolos tradicionales como Modbus no es lo más apropiado, ya que con el tiempo están quedando desfasados. Además hoy día la mayoría de clientes finales, como bases de datos o software HMI cuentan con la capacidad de leer y escribir datos mediante el protocolo OPC UA. Teniendo en cuenta que para cumplir ciertos objetivos del proyecto se requiere realizar una aplicación SCADA, este abre puertas a la utilización de diferentes software [50].

2.4.2.1. Arquitectura

OPC UA, al igual que todas las versiones OPC anteriores, sigue una arquitectura cliente-servidor. En esta arquitectura el cliente realiza peticiones al servidor, quien posee la información, y este las responde. El cliente tiene la capacidad de establecer a que velocidad se sirven los elementos. [Figura 2.10]

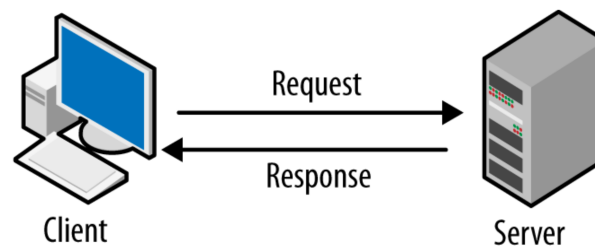


Figura 2.10: Arquitectura Cliente - Servidor [52]

Los clientes se conectan al servidor mediante el endpoint de este último. Un endpoint es una URL que identifica a un servidor, el tipo de transporte y la seguridad con que se comunica. Los servidores OPC soportan una gran cantidad de conexiones de clientes finales, y puede atender a las peticiones de todos ellos simultáneamente [53].

2.4.2.2. Seguridad

Se pueden diferenciar 3 tipos diferentes de seguridad en el protocolo OPC UA:

- Autenticación por usuario/contraseña: el servidor puede requerir a los clientes que se autentifiquen cuando estos se conectan a su endpoint.
- Autenticación mediante intercambio de certificados x509: los certificados son documentos digitales que asocian pares de claves criptográficas con identidades, en este caso el servidor y los clientes. [Figura 2.11]
- Encriptado Basic256Sha256: basado en una serie de algoritmos para encriptado de 256 bits de longitud [50].

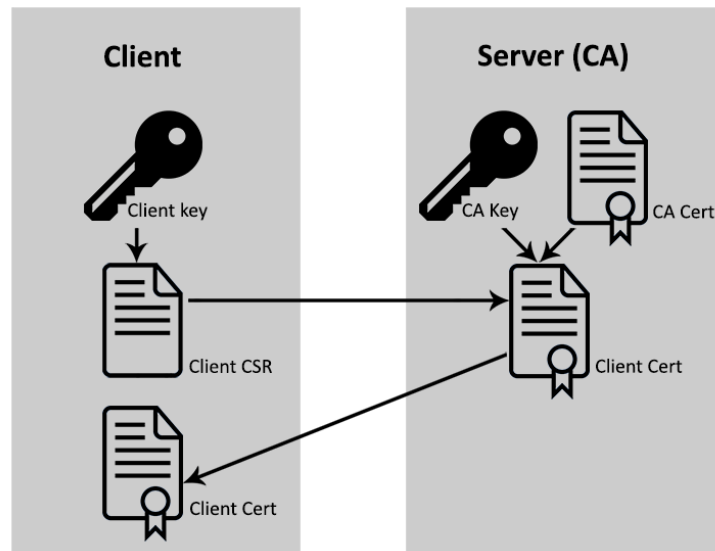


Figura 2.11: Intercambio de certificados entre cliente y servidor [54]

Capítulo 3

Planteamiento y elección de recursos

3.1. Perspectiva general

Para cumplir los objetivos determinados en la fase inicial del proyecto, es conveniente seguir una lógica en el planteamiento de la red y la elección de los recursos utilizados. Por ello, en este apartado se pretende dar contexto del enfoque que se le ha dado al proyecto, así como justificar el hardware y software escogido para cada necesidad del mismo.

Cabe destacar que la realización de este proyecto ha transcurrido durante el periodo de prácticas del estudiante en la empresa Logitek. Es por esto que parte del material ha sido elegido en función de los recursos que la empresa tiene disponibles.

Como ya se ha dejado constancia al inicio del documento, el objetivo principal de este proyecto es la creación de una solución completa de red LoRaWAN integrable en la industria. El esquema [Figura 3.1] y pasos seguidos son los siguientes:

1. En primer lugar, se quiere hacer uso de nodos sensores, los cuales leen los datos que se desean monitorizar en la red. Estos han de ser compatibles con tecnología LoRa, y poder enviar mediante ella sus lecturas a la puerta de enlace. Así mismo deben permitir la comunicación bidireccional, es decir, la recepción de downlinks por parte del gateway.
2. Estos datos son recibidos por el gateway mediante LoRa. Este se encargará de reenviar estos paquetes de datos al servidor de red vía protocolo TCP. Además, reenvía también los mensajes descendentes provenientes de la aplicación.
3. El servidor de red, que es el núcleo de la red LoRaWAN, recibe estos paquetes por TCP. Es aquí donde se encuentra la arquitectura que queremos definir para nuestra red. Se dan de alta todos los dispositivos (nodos, gateways) y aplicaciones (servidores, bases de datos, aplicaciones HMI), así como se

- determina el flujo de comunicación entre estos. En este caso particular, como se utiliza el protocolo MQTT para el envío a las aplicaciones finales, se requiere de un servidor (broker) MQTT externo que almacene los datos publicados por el servidor red.
4. El broker o servidor MQTT es quien serviría los datos a los clientes finales en una arquitectura de red LoRaWAN estandar. Sin embargo, en el esquema de el prototipo de este proyecto hace de puente entre el Network Server y el servidor OPC y Node-RED.
 5. La aplicación Node-RED se suscribe al valor del payload enviado por los dispositivos. Una vez acondicionado, devuelve este valor al broker MQTT.
 6. El servidor OPC, que cuenta con un cliente MQTT, se suscribe a los topics del broker correspondientes a los nodos de la red. Este es el software que permite la integración con el programario comunmente encontrado en la industria. Gracias a su compatibilidad con gran variedad de protocolos (tanto aguas arriba como aguas abajo) este es una pieza clave en esta red. Ofrece la posibilidad de servir los datos a los clientes finales mediante el estándar industrial OPC.
 7. Finalmente, el SCADA es el elemento que ofrece una interfaz visual para que el trabajador pueda monitorizar las lecturas de los sensores, así como controlar y modificar su comportamiento. Desde aquí se leen los datos del servidor OPC mediante el protocolo OPC UA. Cuando se desea enviar un mensaje descendente desde el SCADA a los nodos, este deberá pasar por cada elemento de la red, haciendo uso de los protocolos ya descritos en cada comunicación.

3.2. Hardware

3.2.1. Nodos utilizados

Para este proyecto, se ha hecho uso de los diferentes nodos con los que se contaba en las oficinas de Logitek. Los tres dispositivos pueden encontrarse en el catalogo de nodos LoRa del fabricante Atim. A continuación se detallan características y funcionalidades de cada uno de estos.

3.2.1.1. Sensor de temperatura y humedad

El primer dispositivo se trata de un sensor de temperatura y humedad. El modelo es el ACW/LW8-TH [Figura 3.2]. Se alimenta mediante baterías internas.

Este nodo reporta las lecturas de temperatura y humedad a una puerta de enlace mediante tecnología LoRa. La transmisión de estos paquetes de datos se realiza en intervalos de tiempo regulares, o en alertas

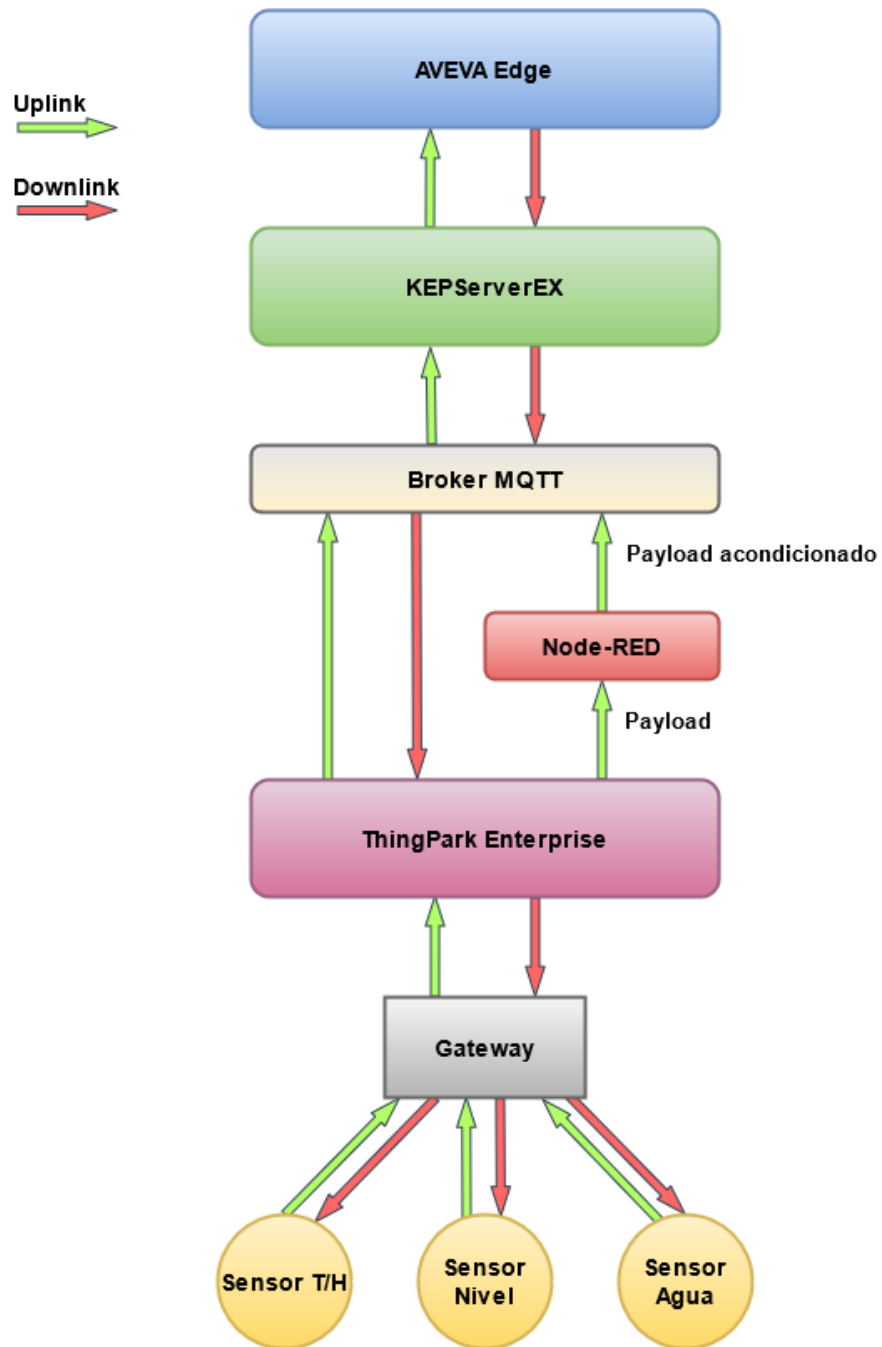


Figura 3.1: Esquema de la red prototipo



Figura 3.2: Vista del nodo ACW/LW8-TH [55]

Cuadro 3.1: Formatos de payload del sensor T/H [3]

Type	Description	Frame format						
		octet 0 (dec)	octet 0 (hex)	octet 1 (hex)	octet 2 (hex)	octet 3 (hex)	octet 4 (hex)	octet 5 (hex)
Keep Alive	Keep-alive frame	1	01	Supply voltage IDLE (mV)		Supply voltage TX (mV)		64
Test	Test frame	5	05	Counter				
TH	Temperature reading frame	23	17	Temperature code		Humidity code		
TH	Alert – reached lower threshold (°C)	15	0F	Temperature code		Humidity code		
TH	Alert – reached upper threshold (°C)	17	11	Temperature code		Humidity code		

cuando el valor leído supera el máximo o mínimo establecido. Es un dispositivo de clase A, por lo que en el momento de enviar los datos deja 2 pequeñas ventanas de tiempo para la recepción de mensajes descendentes. Tiene 2 modos de comportamiento:

- Modo periódico: el envío de datos se realiza de forma periódica, con el intervalo de tiempo definido por el usuario. El valor mínimo de este intervalo es de 10 minutos.
- Modo ahorro de energía: este modo de funcionamiento está orientado a minimizar el consumo de energía y maximizar la duración de esta. Cuando este modo está activo, el dispositivo puede configurarse para enviar paquetes de datos solamente bajo ciertas condiciones, como una variedad notable en el valor de dos lecturas consecutivas o el sobrepasamiento de los valores máximos y mínimos.

En la tabla 3.1 puede observarse el formato de los diferentes tipos de payloads que puede enviar el sensor:

Cabe destacar el tipo más relevante y común para esta red, el mensaje de lectura de temperatura. Como se aprecia en la tabla, todos los payloads están conformados por caracteres hexadecimales. Para estos mensajes en concreto, el payload empieza por el octeto fijo 17, seguido por 4 caracteres que indican un código de temperatura y otros 4 que indican un código de humedad. La conversión a número decimal de estos códigos no representa el valor real obtenido. Sin embargo, este puede obtenerse mediante los siguientes factores de conversión:

$$T(^{\circ}\text{C}) = \frac{\text{CódigoTemperatura} \cdot 175,72}{65536} - 46,85$$

$$H(\%RH) = \frac{\text{CódigoHumedad} \cdot 125}{65536} - 6$$

Así mismo, este sensor soporta la recepción de mensajes descendentes, provenientes tanto del servidor red como de los clientes finales. Estos downlinks deben contener un payload conformado por un número par

Cuadro 3.2: Formato de downlink de periodo de muestreo [3]

Parameter code (Byte 1)	Parameter value (Byte 2)
0x0C	0xYY
0x0D	0xZZ



Figura 3.3: Vista del nodo ACW/LW8-WL [56]

de caracteres hexadecimales, el cual permite realizar ciertos cambios en su configuración. El formato de los diferentes payloads a enviar se detalla en el manual del dispositivo. Como ejemplo más común, en la tabla 3.2 se muestra el formato que ha de tener un downlink para modificar los valores del periodo de muestreo.

Los primeros 2 caracteres tienen valor fijo e indican el código interno del parámetro que se va a modificar. Los 2 siguientes caracteres los determina el usuario, e indican el valor de horas o minutos en que el sensor debe realizar las lecturas de datos [3].

3.2.1.2. Sensor de fuga de agua

El segundo dispositivo utilizado en este proyecto es un sensor de fuga de agua. El modelo es el ACW/LW8-WL. Este sensor se conecta a una red LoRaWAN para permitir la detección de fugas de agua en aplicaciones como suministros de agua y calefacción, y transmite la alerta en tiempo real. Los dos terminales del panel frontal [Figura 3.3] se utilizan para tomar las lecturas, las cuales son un simple booleano que indica si hay presencia o no de agua.

Debido a que está pensado para colocarse en puntos poco accesibles, el dispositivo se conforma de una caja y una antena unidos por un cable de 4m de largo. El módulo se alimenta internamente con una batería de litio y tiene una clasificación de sellado IP68 para soportar condiciones extremas.

A diferencia del sensor visto anteriormente, el ACW/LW8-WL tiene un uso y configuración muy limitados. Este comprueba la presencia de agua cada 10 minutos. El mensaje ascendente es transmitido si, y solo si, hay una variación en la lectura del sensor. Este cambio en el valor de la lectura hace que se transmitan 5



Figura 3.4: Vista del nodo ACW/LW8-LVL [57]

paquetes de datos LoRa para asegurar que la información es correctamente recibida y procesada. En estos paquetes, el último dígito hexadecimal corresponde al valor de la lectura, 1 si se ha detectado agua y 0 en el caso contrario. Los mensajes descendentes por lo contrario, no son soportados por el dispositivo. Es por esto que no admite cambios en su configuración o comportamiento [4].

3.2.1.3. Sensor de nivel

El último nodo que conforma la red LoRaWAN de este proyecto es un sensor de nivel. El modelo es el ACW/LW8-LVL [Figura 3.4], destinado a aplicaciones como la medición del nivel de llenado de tanques, contenedores o silos.

Este dispositivo, se instala sobre la parte superior del recipiente a medir, y realiza lecturas periódicas del nivel de llenado de este en centímetros teniendo en cuenta la configuración establecida. El intervalo de envío de mensajes de medición puede ser configurado desde un mínimo de 1 minuto a un máximo de 255 horas. Además de esto, puede establecerse que se envíe más de una medición en un mismo paquete de datos. De este modo si se determina un periodo de 12 minutos y 4 muestras, la lectura del nivel se hará cada 3 minutos. Su rango de medición es entre 0.20 y 5.50m. Al igual que el sensor de temperatura y humedad, el ACW/LW8-LVL es altamente configurable mediante el software de Atim o el envío de downlinks.

Comparandolo con los anteriormente vistos, este sensor es el que cuenta con más capacidad para enviar diferentes tipos de uplinks. A continuación se detalla el formato del más común y útil, el mensaje ascendente de medición. Este cuenta con dos bytes dedicados a informar de los parámetros del mensaje. El primero, el byte de encabezado, contiene los datos que constan en la tabla, entre ellos el número de muestras que contiene el mensaje y el histórico de muestras. [Cuadro 3.3]

La configuración del histórico permite el envío de muestras pasadas en un nuevo mensaje. Por ejemplo, si se establece el valor histórico a 3, en los mensajes ascendentes de medición constará la muestra de distancia

Cuadro 3.3: Byte 1 del uplink de lectura del ACW/LW8-LVL [5]

Byte 1 - Header							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
New generation = 1	Timestamp	Measurement frame = 1	History depth (-1) Max: 4		Number of samplings (-1) Max: 8		

Cuadro 3.4: Byte 2 del uplink de lectura del ACW/LW8-LVL [5]

Channel header							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Reserved = 0		Channel's number		Type of measurement			

tomada actualmente, así como las tomadas en los 2 anteriores mensajes.

El segundo byte informativo, el de encabezado de canal, muestra el número de canal y el tipo de medida que se está enviando. [Cuadro 3.4]

Tras estos, le siguen los bytes con el valor de la distancia medida. El tamaño total del mensaje está determinado por el número de muestras que este contiene. Para cada muestra, se necesitan 2 bytes (4 caracteres hexadecimales). En el cuadro 3.5 se puede observar un ejemplo de mensaje ascendente de medición, con un total de 2 muestras tomadas.

Los bytes 2 y 3, aparecen en aquellos mensajes con más de una muestra leída, e indican el período de tiempo en que se han tomado dichas muestras. Volviendo al ejemplo, estos tienen un valor de 0x003C (60 en formato decimal), es decir, 60 minutos para tomar ambas muestras. Como las muestras son tomadas en períodos idénticos, estas se han leído cada 30 minutos. Finalmente los bytes 5 y 6 del mensaje corresponden a la primera muestra tomada (0x0122, 290 centímetros), mientras que los bytes 7 y 8 corresponden a la segunda (0x0175, 373 centímetros).

Como ya se ha mencionado anteriormente, este nodo soporta mensajes descendentes para modificar su configuración. Algunos de los valores modificables son el offset y los umbrales para el envío de alertas. La siguiente tabla corresponde al formato de un downlink para la modificación de los parámetros de los

Cuadro 3.5: Bytes de distancia medida del ACW/LW8-LVL [5]

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0xA1 (new generation measurement frame, no history, 2 samples)	0x003C Radio transfer period		0x07 (channel 0, measurement type: distance)	0x01	0x22	0x01	0x75

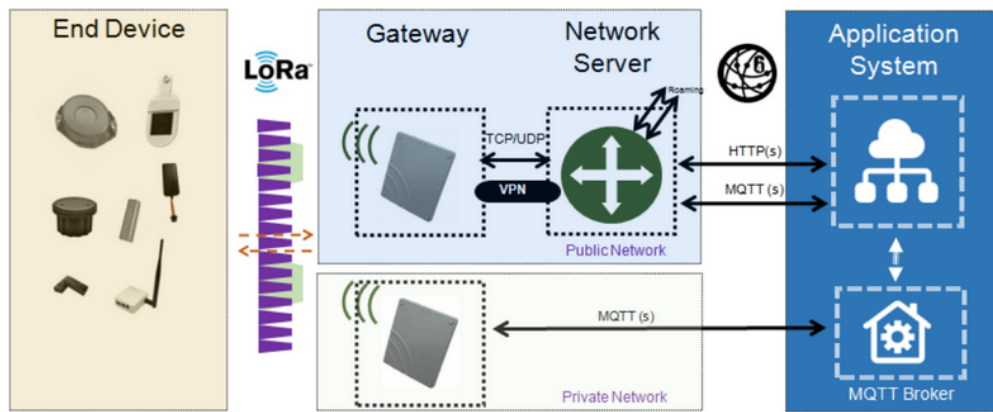


Figura 3.5: Diferentes modos del Gateway ODU Macro [6]

paquetes de datos.

Para el byte 4, los dos bits YY corresponden al histórico deseado, con un valor máximo de 4. Los bits ZZZ corresponden al número de muestras por mensaje. Los bytes 5 y 6 corresponden al valor en minutos del periodo entre el envío de mensajes [5].

3.2.2. Gateway

El modelo de puerta de enlace utilizado en este proyecto es el ODU Macro V1.5 - 8 channels, del fabricante Gemtek. Este es un dispositivo orientado a funcionar en exteriores, es por esto que su caja está sellada y preparada para soportar condiciones desfavorables. Gemtek comercializa este gateway con 2 configuraciones diferentes según el uso que se le vaya a dar [Figura 3.5]:

- Modo LoRaWAN: con servidor de red integrado.
- Modo reenviador de paquetes: con software propietario de diferentes fabricantes para trabajar con servidores de red específicos.

En nuestro caso, el dispositivo se nos fue distribuido por la compañía Actility, por lo que venía con un software de fábrica (LRR Software) para comunicar específicamente con su Network Server, ThingPark Enterprise.

3.2.2.1. Características principales

Este modelo particular, no cuenta con tecnología WiFi, por lo que debe conectarse a la red de forma alámbrica mediante ethernet. El esquema de conexión a esta es el mostrado en la figura 3.6.

El software propietario de Actility, cuyo funcionamiento se detallará en próximos apartados, cuenta con una interfaz accesible al usuario mediante conexión SSH (Secure SHell).

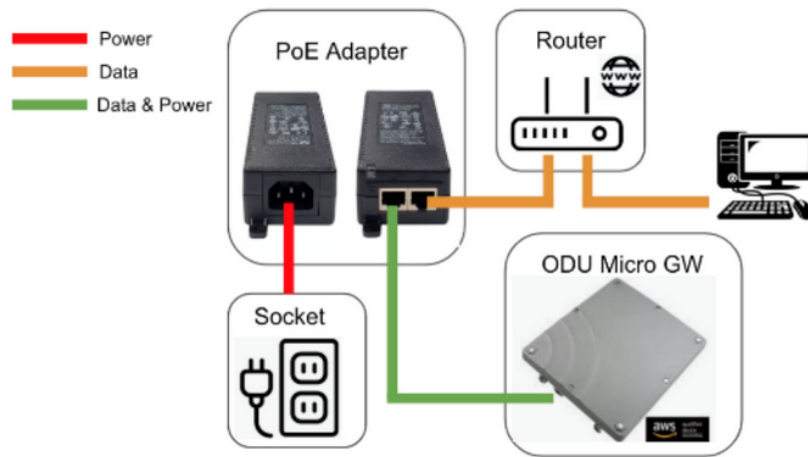


Figura 3.6: Esquema de conexión del Gateway ODU Macro [58]



Figura 3.7: Puertos físicos del Gateway ODU Macro [59]

En cada mensaje ascendente recibido de los nodos, el gateway se encarga de añadir una serie de parámetros de red.

Los parámetros recibidos en los downlinks en cambio vienen determinados por el servidor de aplicaciones.

3.2.2.2. Puertos físicos

En la figura 3.7, se indican los diferentes puertos con los que cuenta el equipo:

- Los puertos 1 y 2 permiten la conexión de antenas de radiofrecuencia.
- El punto 3 corresponde a una tapa transparente de rosca. Una vez retirada, se accede a una ranura que soporta tarjetas mini SIM 3G/4G.
- El puerto 4 habilita el acoplado de una antena GPS de 1.5GHz, para la localización del gateway.

- El puerto 5 corresponde a una entrada ethernet, la cual permite la conexión del equipo a la red y a su vez ser alimentado.
- El puerto 6 permite la conexión de una antena 3G/4G [59].

3.2.2.3. Funcionalidad

Como ya se ha indicado en la introducción de este mismo apartado, la funcionalidad principal de esta puerta de enlace es la de reenviar los paquetes LoRa recibidos al servidor red ThingPark Enterprise.

El ODU Macro V1.5 se encarga también de verificar la integridad de los paquetes de datos recibidos de los nodos. Para ello, analiza el CRC de cada mensaje, y solo envía al Network Server aquellos paquetes que ha captado íntegramente [6].

3.3. Software

El software elegido para conformar la red LoRaWAN privada de este proyecto se ha determinado teniendo en cuenta dos factores:

- Las limitaciones impuestas por el Gateway: Actility comercializa el ODU Macro V1.5 con su software propietario instalado. Esto limita las comunicaciones del equipo con servidores red de otros fabricantes. A cambio, se le ofrece al usuario una configuración fácil y amigable, la cual no requiere mucho tiempo para dar de alta la puerta de enlace en ThingPark Enterprise.
- El catálogo de programario que ofrece la empresa Logitek. Entre muchos otros softwares, Logitek distribuye programas orientados a satisfacer necesidades en el ámbito de la monitorización y las comunicaciones industriales. Estos programas permiten la integración de la solución LoRaWAN de Actility en cualquier planta industrial actual.

3.3.1. Servidor red ThingPark Enterprise

Como ya se ha mencionado en anteriores apartados, el servidor de red implementado en la red LoRaWAN de este proyecto es ThingPark Enterprise.

ThingPark Enterprise es una plataforma IoT optimizada para gestionar una red LoRaWAN a una escala empresarial. Este puede utilizarse tanto en un navegador web (software como servicio) como en una instancia de programa ejecutándose en las instalaciones del cliente. Para este proyecto, se ha utilizado ThingPark Enterprise como servicio, el cual requiere conexión a Internet entre las puertas de enlace LoRa y las redes núcleo de ThingPark basadas en la nube.

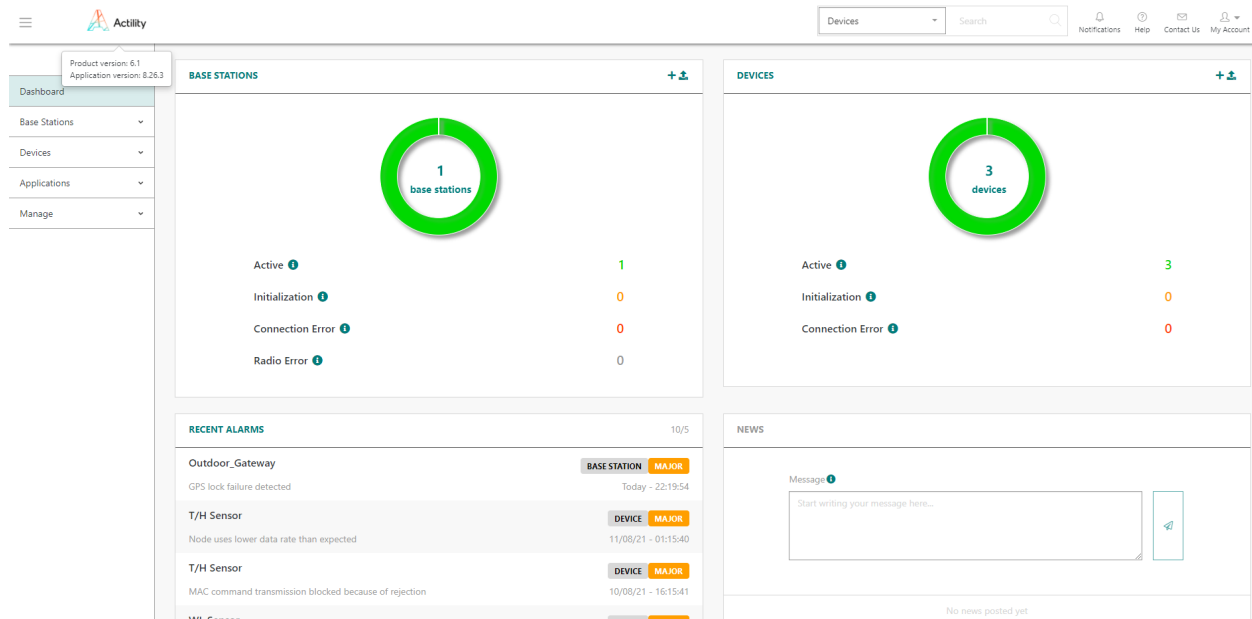


Figura 3.8: Interfaz de ThingPark Enterprise [60]

3.3.1.1. Interfaz

En la imagen 3.8 se muestra la página principal de la interfaz de ThingPark Enterprise como servicio.

A la izquierda de la pantalla, se puede observar el menú de la plataforma, el cual permite acceder a las diferentes secciones correspondientes a:

- Tablero
- Gateways
- Dispositivos (nodos)
- Aplicaciones
- Gestionar

La página principal o tablero le ofrece al usuario una representación general de la magnitud y el estado de su red LoRaWAN. En ella se aprecian un par de gráficas tipo ‘donut’, las cuales indican el número de puertas de enlace y nodos dados de alta en la red, así como su estado operativo o la presencia de errores. En la parte superior derecha de cada uno de estos, se encuentra un acceso directo al menú de adición de dispositivos.

Debajo de estos, se encuentra un panel que muestra las alarmas más recientes generadas en la red. En ellas se indica el dispositivo implicado y el grado de gravedad. A su derecha está la sección de noticias. Esta simplemente permite publicar mensajes en el tablero, para que otros usuarios de esta misma cuenta puedan leerlos.

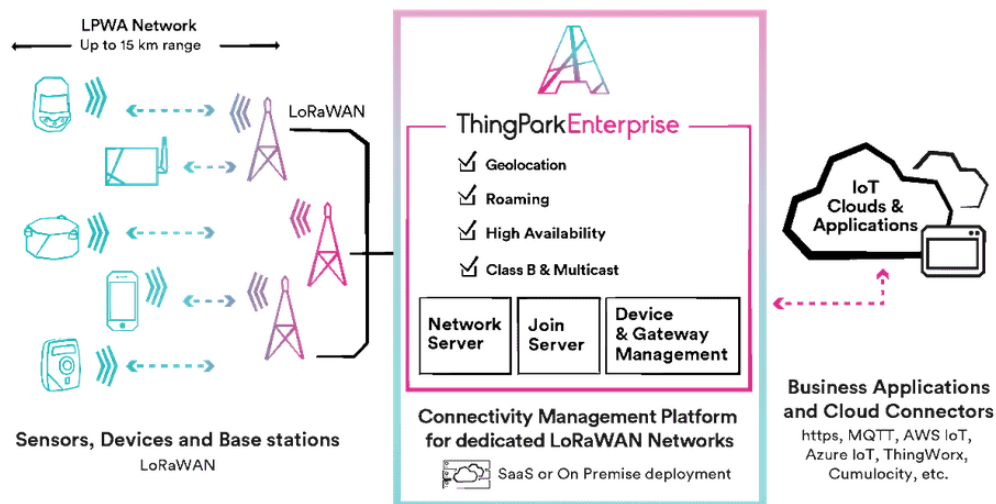


Figura 3.9: ThingPark Enterprise en una red LoRaWAN estándar [7]

Por último, en el inferior de la pantalla se muestra una tabla resumen de las aplicaciones configuradas en la red. De estas se indica el tipo, nombre, identificador y número de nodos asociados a cada una de ellas [61].

3.3.1.2. Funcionalidad

ThingPark Enterprise ofrece a los administradores de la red LoRaWAN las siguientes posibilidades:

- Formar la infraestructura de red creando, gestionando y eliminando puertas de enlace LoRa.
- Dar de alta nuevos nodos en la red y gestionarlos.
- Monitorizar las operaciones de la red mediante el uso de tablas, alarmas y información de estado.
- Control del flujo de datos a los application servers o a populares plataformas IoT en la nube.

3.3.1.3. Funcionamiento

La imagen 3.9 muestra la ubicación de ThingPark Enterprise dentro de la estructura de una red LoRaWAN estándar.

El funcionamiento de la plataforma ThingPark Enterprise se basa en la interfaz de modo Túnel. Esta interfaz actúa como transmisora de mensajes entre los Application Servers y los servidores ThingPark (Long Range Controller servers), los cuales implementan la capa de red MAC (Medium Access Control). La capa MAC es el conjunto de procesos y protocolos mediante los cuales diferentes dispositivos se ponen de acuerdo para compartir un medio físico de transmisión.

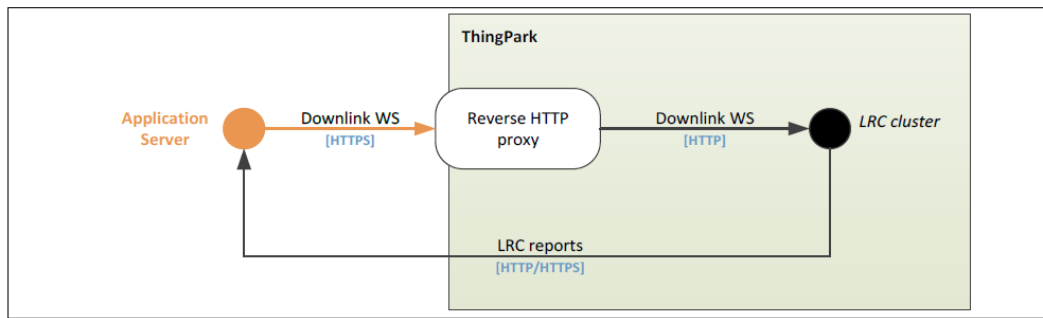


Figura 3.10: Interfaz Túnel entre ThingPark Enterprise y los Application Servers [62]

Esta interfaz [Figura 3.10] reenvía los mensajes ascendentes enviados por el Gateway, cuyo contenido consta del dato bruto leído por el sensor (payload) más los metadatos de red generados por la puerta de enlace. Estos uplinks son enviados a uno o más servidores de aplicaciones asociados a la dirección del correspondiente nodo (DevEUI). Como ThingPark Enterprise soporta la comunicación bidireccional, los servidores de aplicaciones también pueden hacer solicitudes a uno de los nodos del controlador de largo alcance para enviar mensajes descendentes al nodo determinado [62].

3.3.2. Node-RED

3.3.2.1. Funcionalidad

Node-RED es una herramienta de desarrollo basada en programación visual de flujos para conectar dispositivos, APIs y servicios online, principalmente orientada a aplicaciones IoT. Esta proporciona un editor de flujos accesible a través de cualquier navegador de Internet, el cual permite crear funciones en lenguaje de programación JavaScript. Cada una de estas funciones está visualmente representada por un bloque (nodo). Node-RED ofrece nodos predeterminados con funcionalidades comunes, así como la posibilidad de crear nodos con funciones propias [8].

3.3.2.2. Interfaz

En la figura 3.11, se muestra la ventana del editor de flujos de Node-RED.

A grandes rasgos, este cuenta con 4 diferentes componentes:

- Encabezado: es la sección superior del editor. Contiene el botón de despliegue del flujo, el menú principal y, si la autenticación está habilitada, el menú de usuarios.
- Paleta: localizada en la parte izquierda de la ventana, la paleta contiene todos los nodos predeterminados de Node-RED. Desde esta, el usuario puede arrastrar los nodos y colocarlos en el centro para crear flujos.

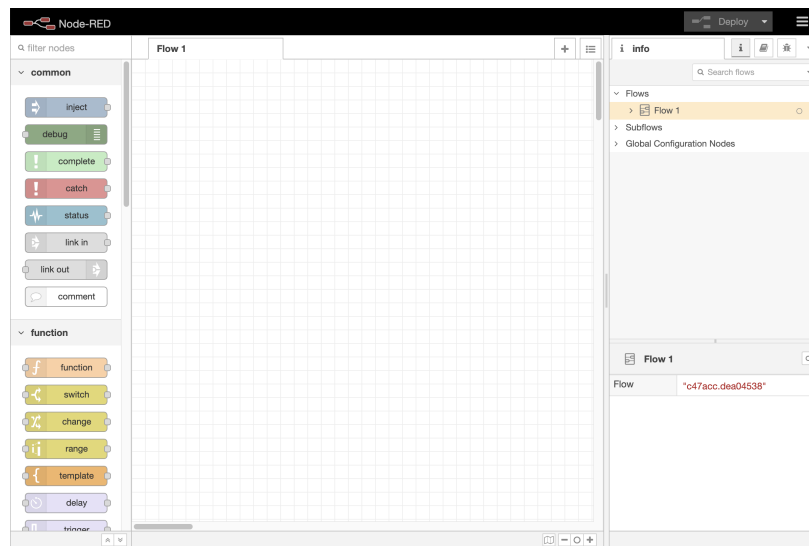


Figura 3.11: Interfaz de Node-RED [63]

- Espacio de trabajo: situado en el centro del editor, es el area en que el usuario puede crear flujos.
- Barra lateral: es la columna derecha de la ventana del editor. En ella, puede accederse a los menús Información, Debug, Nodos de Configuración y Datos de Contexto [63].

3.3.3. Servidor OPC KEPServerEX

En términos generales, un servidor OPC es un software que recoge datos de los dispositivos de campo (PLCs, módulos de entradas y salidas, controladores, etc.) u otras fuentes utilizando sus protocolos de comunicación nativos, y los pone al servicio de los clientes finales (SCADAs, interfaces humano-máquina, bases de datos, etc.) mediante una o más especificaciones definidas por la OPC Foundation.

Este tipo de programario permite a las industrias interconectar equipos y softwares de gran variedad de fabricantes, sin tener que preocuparse de compatibilidades, conexiones y protocolos.

El servidor OPC utilizado en este proyecto es KEPServerEX. Este servidor es una solución de conectividad que permite leer cualquier dato proveniente de un proceso industrial y servirlo mediante el estándar OPC UA, entre otros protocolos, a los clientes finales conectados.

3.3.3.1. Características principales

KEPServerEX cuenta con multitud de drivers capaces de comunicar aguas abajo con dispositivos y fuentes comunmente utilizadas en la industria mediante sus protocolos propietarios. Algunos de los drivers más utilizados son el cliente OPC UA, Siemens TCP/IP Ethernet y Modbus.

Para comunicar aguas arriba con los clientes finales, este servidor OPC dispone de varias interfaces, como pueden ser OPC, ODBC (Open DataBase Connectivity) y REST (Representational State Transfer).

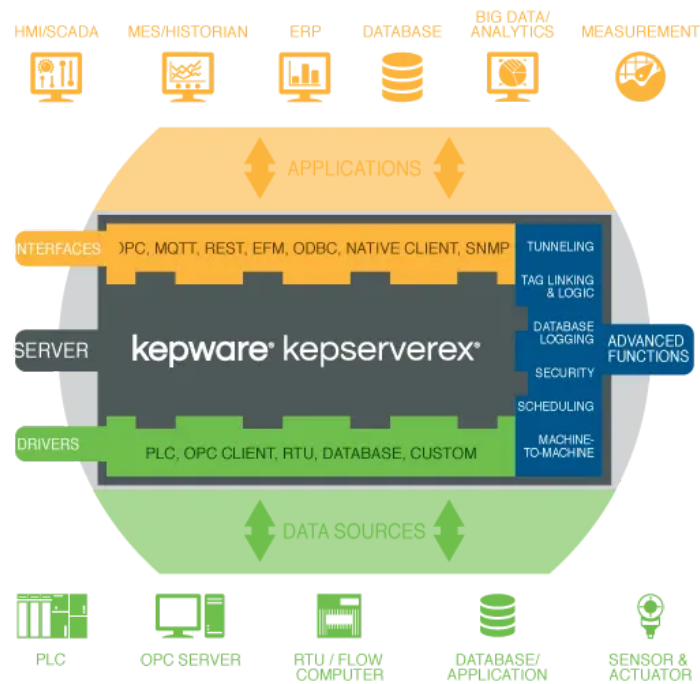


Figura 3.12: Comunicaciones establecidas por KEPServerEX [9]

[Figura 3.12]

KEPServerEX encaja con los objetivos del presente proyecto debido a la disponibilidad del driver cliente MQTT y la interfaz OPC UA. Además de esto, ofrece amplias opciones de configuración para adaptar las comunicaciones a las necesidades del sistema y el usuario [9].

3.3.3.2. Interfaz

La imagen 3.13 muestra la interfaz del KEPServerEX Configuration, el proceso donde se crea y modifica la arquitectura y configuración del proyecto.

En la columna de la izquierda se encuentran las funcionalidades del servidor. En la sección Connectivity se pueden crear y modificar los distintos canales (protocolo de comunicación, dirección del dispositivo, configuración de la sesión) y dispositivos (configuración de parámetros, tasas de lectura de datos, timeouts). En resumidas cuentas, en esta sección se gestionan las comunicaciones aguas abajo con los dispositivos de campo. Es aquí donde se debe configurar un canal para comunicar con el conector MQTT del servidor web ThingPark Enterprise. Debajo de esta, hay accesibles otros plug-ins o herramientas orientados a usos diferenciados.

En la parte central de la pantalla, generalmente se muestran los tags (variables) de los dispositivos. Estos pueden ser generados automáticamente dependiendo del driver o definidos manualmente por el usuario.

Por último, en el inferior de la pantalla se dispone de un registro de eventos. En este se muestra in-

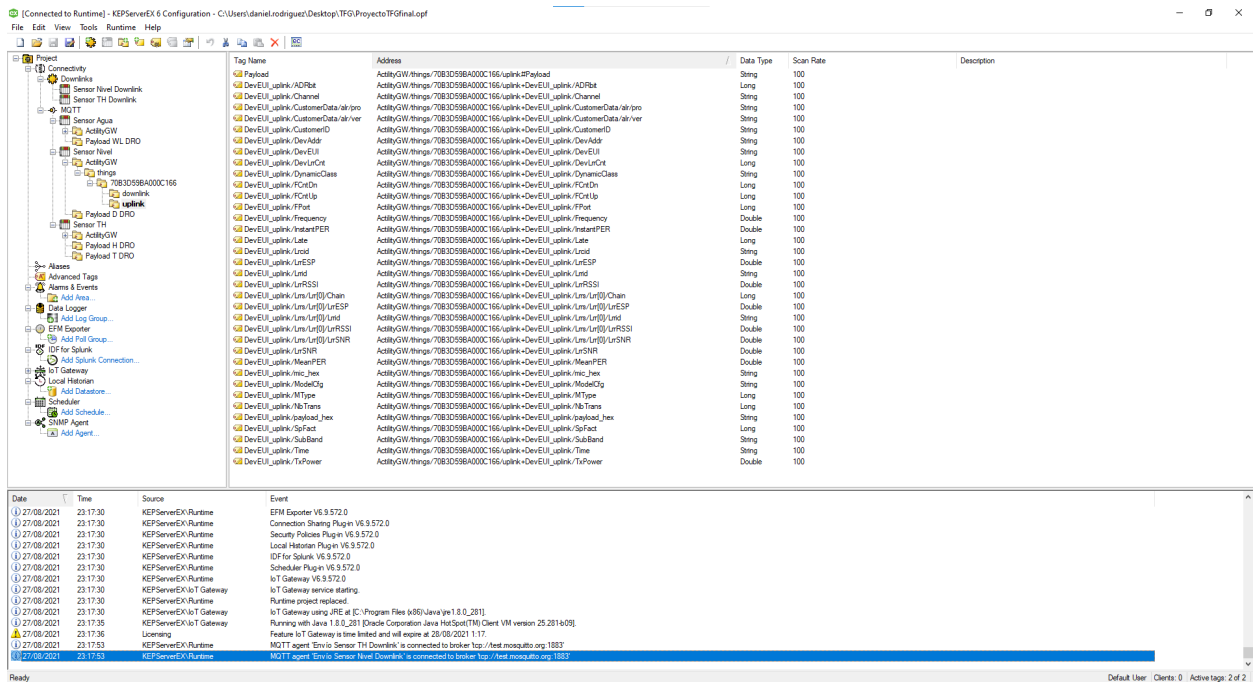


Figura 3.13: Interfaz de KEPServerEX Configuration

formación sobre los procesos que está ejecutando el servidor, así como avisos y errores referentes a las comunicaciones establecidas.

Respecto a las comunicaciones aguas arriba, estas se gestionan en dos ubicaciones de KEPServerEX:

- **Propiedades del proyecto:** en esta ventana [Figura 3.14], se determina mediante qué interfaces es accesible el proyecto actual. Así mismo, pueden configurarse diferentes parámetros referentes a la conexión y la seguridad.
- **KEPServerEX Administration:** al igual que KEPServerEX Configuration, este es otro proceso del servidor en el cual se configuran parámetros generales del software. Entre otros, se encuentra el menú OPC UA Configuration Manager [Figura 3.15], el cual permite añadir, editar y eliminar clientes y servidores fiables, así como gestionar sus respectivos certificados.

Una vez un cliente se conecta al servidor mediante alguna de sus interfaces, puede leer y escribir en todos los tags determinados en la configuración.

3.3.4. Cliente SCADA

Un cliente SCADA es una herramienta de automatización y control industrial, utilizada comunmente para controlar, supervisar, recopilar y analizar datos. Su elemento más relevante es la HMI. Esta es la interfaz que conecta al usuario con la máquina, presentando los datos del proceso mediante un sistema de monitoreo. Además, permite al usuario controlar y realizar cambios sobre las variables del sistema.

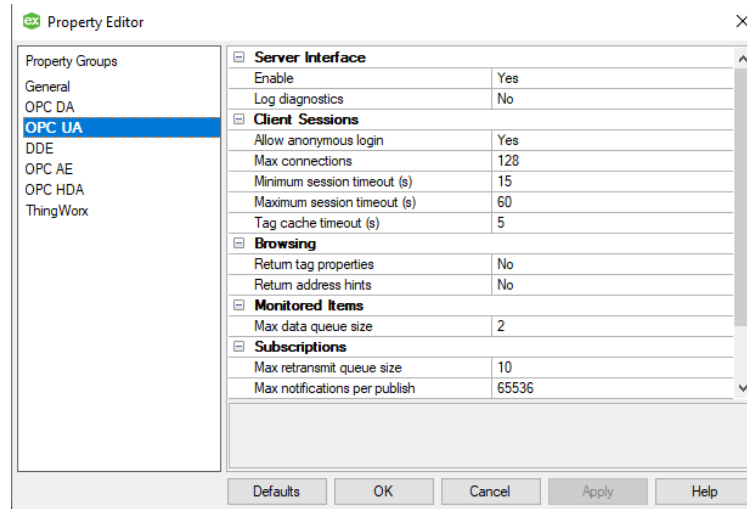


Figura 3.14: Propiedades del proyecto de KEPServerEX

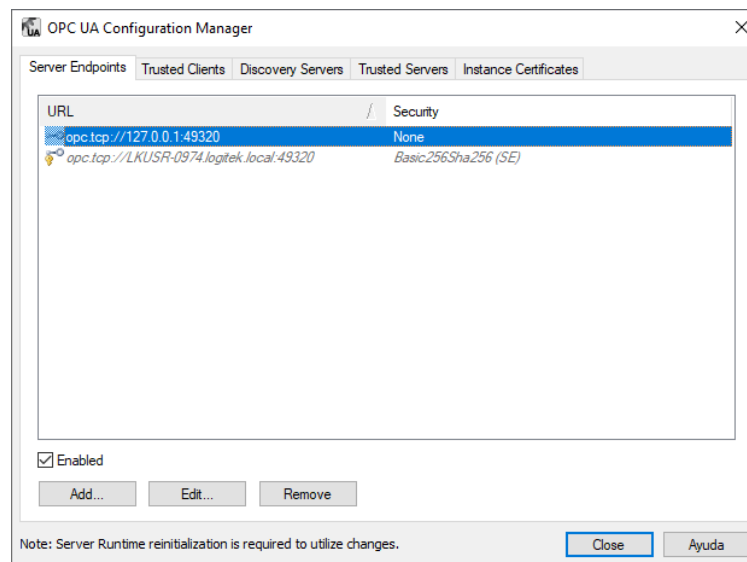


Figura 3.15: KEPServerEX OPC UA Configuration Manager

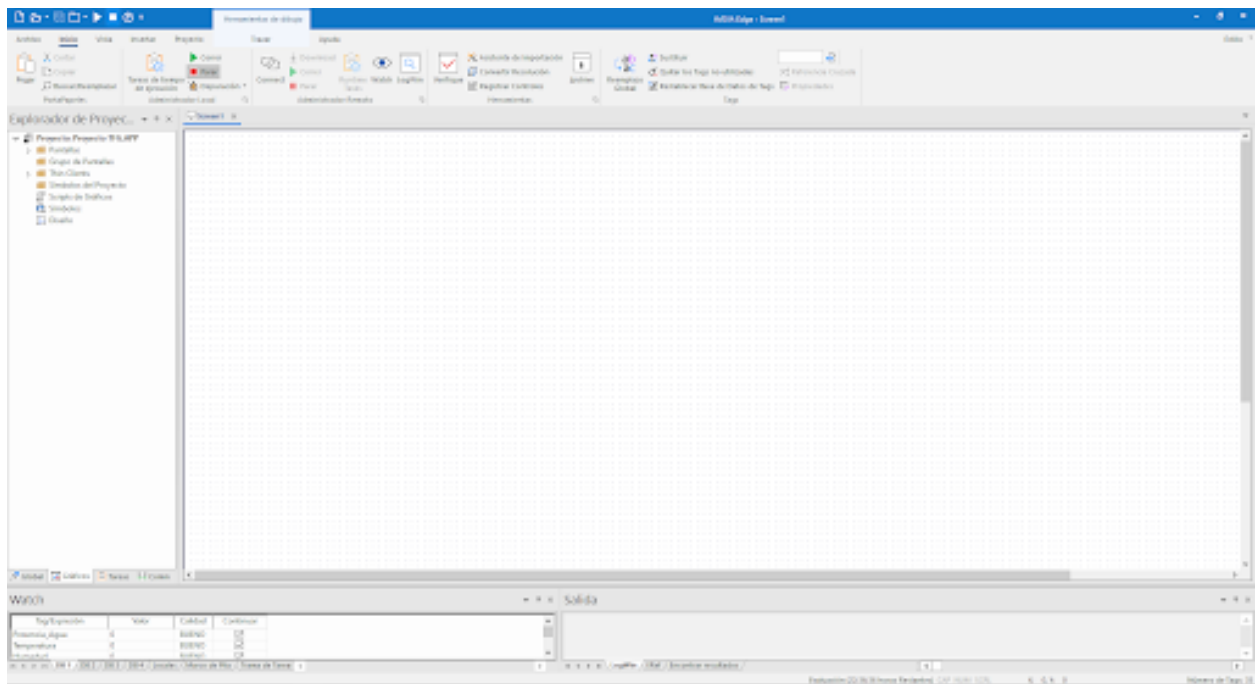


Figura 3.16: Interfaz de Aveva Edge 2020 Studio

El software SCADA elegido para este proyecto es AVEVA Edge. Este cliente pone las funcionalidades habitualmente encontradas en aplicaciones de monitorización dentro de una interfaz amigable al usuario. Además, pone a disposición una amplia biblioteca de iconos símbolos que ayudan a ahorrar tiempo en el diseño de los elementos gráficos [64].

3.3.4.1. Características

AVEVA Edge cuenta con más de 250 drivers que permiten una comunicación fluida entre el HMI y todos los principales fabricantes de PLC y controladores. Entre estos, se incluye el protocolo OPC UA. Además de su gran polivalencia en comunicaciones aguas abajo, este programa permite lanzar aplicaciones SCADA en múltiples plataformas como ordenadores industriales, portátiles, teléfonos y tabletas, o la nube [10].

3.3.4.2. Interfaz

En la figura 3.16 se muestra la interfaz de AVEVA Edge 2020 Studio, el proceso en el cual se diseña y programa la aplicación SCADA.

En la parte superior de la ventana se encuentra la denominada ‘cinta’. Esta combina numerosos menus y barras de herramientas de previas versiones del software en una única interfaz amigable al usuario. Casi todos los comandos están disponibles en la cinta, organizados en diferentes pestañas y grupos acorde a su uso general.

En la parte izquierda, se dispone del Explorador de Proyectos. Este menú organiza todas las pantallas,

hojas de datos y otros ítems que componen el proyecto y los presenta en una extensible vista de árbol. Este cuenta con 4 pestañas: Global, Gráficos, Tareas y Comunicaciones.

La parte central de la pantalla corresponde al Editor de Pantallas y Hojas de trabajo. Es aquí donde se edita y se trabaja con la mayoría de elementos del proyecto.

Por último, en la parte inferior se dispone de las ventanas 'Watch' y Salida. En la primera, se permite al usuario visualizar el valor de los tags del proyecto que se deseen, así como modificarlo. A su derecha, en la ventana Salida se muestran los mensajes referentes a la ejecución de la aplicación o el programa.

Capítulo 4

Desarrollo de la red

En este apartado de la memoria, se pretende definir el procedimiento paso a paso seguido para realizar el montaje de la red privada LoRaWAN. La disposición de los subapartados viene determinado por el orden seguido a la hora de ir ensamblando la red.

4.1. Instalación física de los equipos

4.1.1. Instalación del Gateway

En la caja del equipo, pueden encontrarse los siguientes elementos:

- El gateway ODU Macro V1.5 - 8 channels.
- Un transformador con cable para conectar a la corriente.
- Una antena tipo dipolo, para transmitir o recibir ondas de radiofrecuencia.
- Una antena GPS.
- Un kit de montaje para acoplar el dispositivo en superficies específicas.

Se procede a la instalación del equipo siguiendo el esquema mostrado en la figura 3.6, extraído de la documentación del Gateway.

A un lado del transformador, se conecta el cable que va a la corriente para alimentar el equipo. En el otro, se dispone de dos puertos Ethernet. El primero se utiliza para comunicar con la puerta de enlace, así como alimentarla mediante el método 'Power over Ethernet'. El segundo puerto Ethernet del transformador se utiliza para conectar el equipo al router de red local (en este caso mi router de red doméstica). Una vez aplicadas estas conexiones, el transformador puede conectarse a la corriente eléctrica y el Gateway pasa a ser un dispositivo más en la red local.

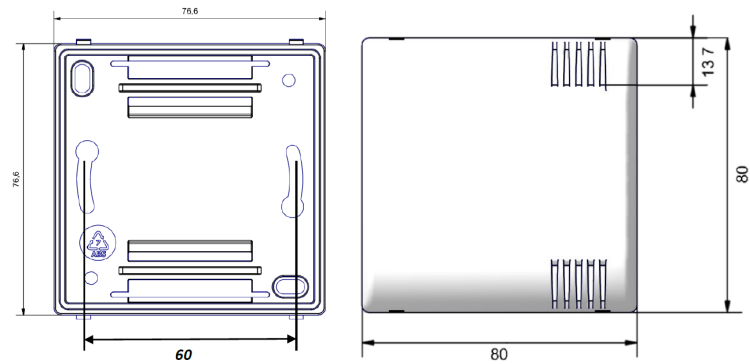


Figura 4.1: Esquema acotado del nodo ACW/LW8-TH [3]

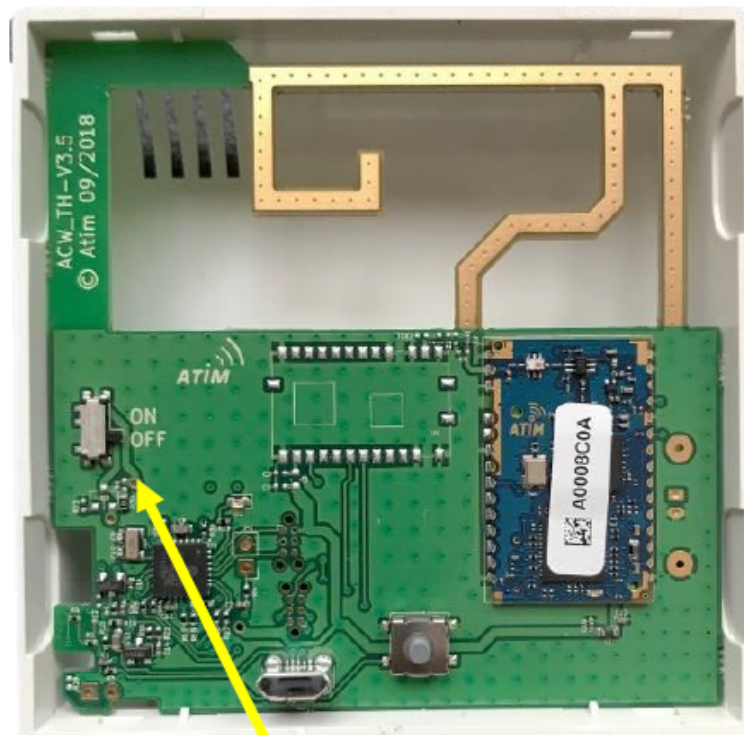


Figura 4.2: Placa interior del nodo ACW/LW8-TH [3]

4.1.2. Instalación del sensor de temperatura y humedad

En la figura 4.1 se muestra un esquema de la caja del ACW/LW8-TH.

Esta viene con agujeros en la parte posterior para ser montado con tornillos en una pared. Se recomienda montarlo en una superficie vertical o acoplado a una pared a una altura mínima de 2m. Esta tapa posterior puede extraerse para acceder a la placa y los circuitos del dispositivo. [Figura 4.2]

A la izquierda de la placa, se encuentra el interruptor para encender el dispositivo. En la parte inferior se dispone de un puerto micro USB (Universal Serial Bus), el cual puede utilizarse para conectar el sensor al ordenador y configurarlo mediante el software ACW Configurator. Este puerto también puede utilizarse para recargar la batería, colocada detrás de la placa. En la parte derecha del ensamblado está el transmissor LoRa, cuyo indicador LED se ilumina con luz verde cada vez que se envía un paquete de datos.

Una vez se acciona el interruptor de despertado, el dispositivo envía 3 mensajes iniciales y posteriormente pasa a enviar sus lecturas cada 10 minutos (valor por defecto), a la espera de ser recibidas por algún Gateway [3].

4.1.3. Instalación del sensor de fuga de agua

Como ya se ha comentado anteriormente, este es el nodo más sencillo en todos sus aspectos, también en su instalación. Al tratarse de una caja perfectamente sellada, esta no puede abrirse ni cuenta con ninguna clase de botones/interruptores para su configuración. Únicamente pueden apreciarse sus 2 terminales conductores, los cuales realizan lecturas para detectar o no la presencia de agua.

El dispositivo puede montarse a una superficie mediante tornillos utilizando los agujeros presentes en la parte inferior de su caja. Una vez acoplado en el lugar a monitorizar, debe colocarse la antena en un lugar lo más alto posible y libre de obstrucciones ambientales, como podría ser una caja eléctrica. Finalmente, para despertar el dispositivo debe retirar el imán que viene pegado en la parte posterior de la caja. El dispositivo comenzará a realizar lecturas periódicas cada 10 minutos a partir de este momento. Enviará la lectura inicial y tras esta solo enviará las lecturas que hayan supuesto un cambio de valor. Si se desea parar el dispositivo se debe acoplar este o cualquier otro imán en el mismo punto [4].

4.1.4. Instalación del sensor de nivel

Para proceder a la instalación del dispositivo, primeramente es recomendable conectar la antena radiofrecuencia en su correspondiente entrada.

Antes de acoblar el dispositivo en el lugar a monitorizar, es necesario despertarlo y establecer la configuración determinada. Para realizarlo es conveniente quitar la tapa superior. De este modo se puede acceder a su placa y al puerto micro USB integrado en ella. Mediante este puede conectarse al ordenador y configurarse con el software propietario ACW Configurator.

También con la tapa retirada, se procede a despertar el dispositivo. Para ello, acercar un imán en la zona resaltada de la imagen y mantenerlo pegado a la caja durante 6 segundos. [Figura 4.3]

Durante este periodo, el led parpadea con luz blanca. Al finalizar los 6 segundos, el led parpadea con luz verde una vez. Es entonces cuando debe retirarse el imán. Hecho esto el sensor está despierto y pasa a la fase de emparejamiento, la cual se indica con luz lila en el led indicador de estado. Durante esta fase el sensor busca Gateways cercanos y testea su comunicación, pudiendo indicar 3 resultados diferentes en el led según la calidad de la red:

- Luz verde, para una buena calidad de red.

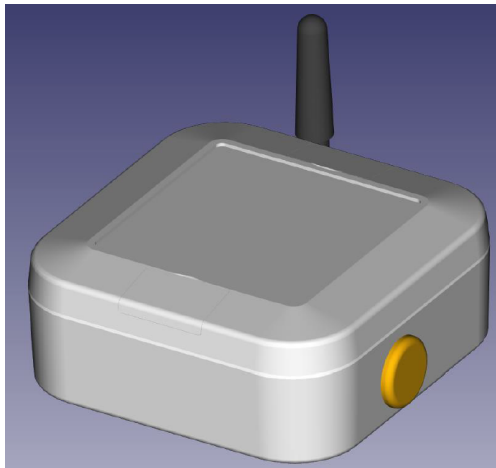


Figura 4.3: Imán de activación del nodo ACW/LW8-WL [5]

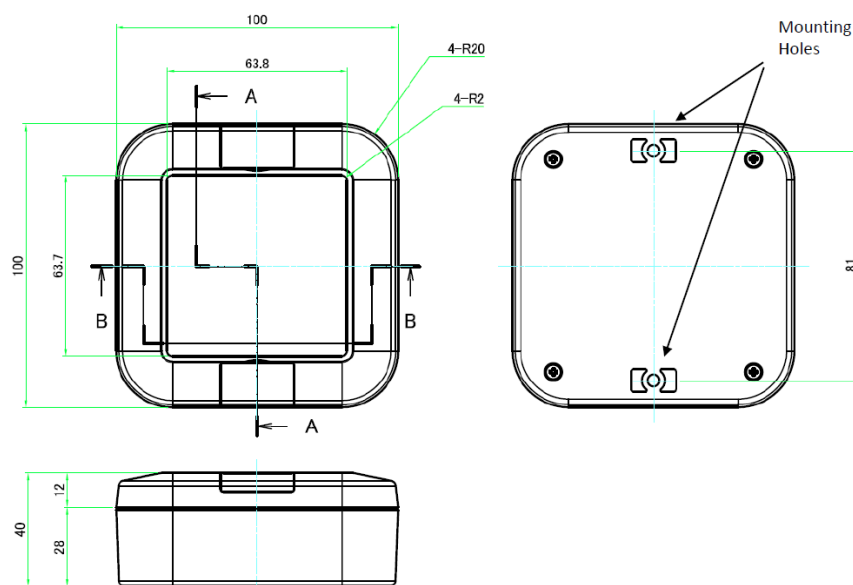


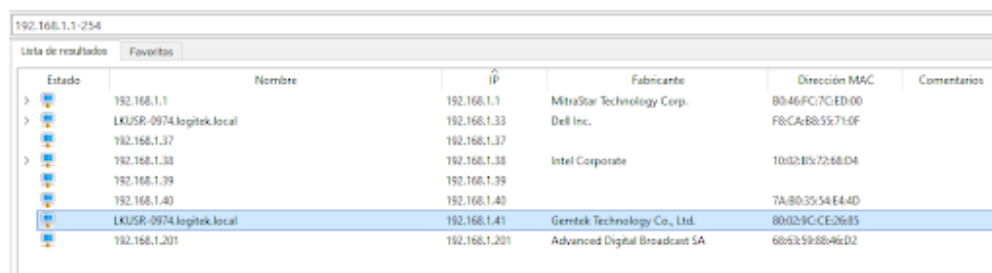
Figura 4.4: Esquema acotado del nodo ACW/LW8-WL [5]

- Luz amarilla, para una baja calidad de red.
- Luz naranja, para una mala calidad de red.
- Luz blanca, si no se ha detectado información de ninguna red. Una vez indicado el resultado el dispositivo pasa a estar en modo operativo, realizando las lecturas determinadas en su configuración.

Si se desea detener el funcionamiento del sensor, se debe acercar el imán en el mismo punto durante otros 6 segundos. Durante este periodo, el led indicador de estado parpadeará con el color de su estado actual. Finalizado el tiempo, la luz del led se vuelve blanca y se desvanece poco después, indicando que el dispositivo ha sido puesto en estado standby. Hecho esto puede retirarse el imán.

En la figura 4.4 se muestra un esquema con las dimensiones de la caja.

Esta puede montarse en el lugar de monitorización de tres formas distintas:



Estado	Nombre	IP	Fabricante	Dirección MAC	Comentarios
>	192.168.1.1	192.168.1.1	MitraStar Technology Corp.	B0:46:FC:7C:ED:00	
>	LKUSR-0974.logitek.local	192.168.1.33	Dell Inc.	F8:CA:88:52:71:0F	
>	192.168.1.37	192.168.1.37			
>	192.168.1.38	192.168.1.38	Intel Corporate	10:02:85:72:88:D4	
>	192.168.1.39	192.168.1.39			
>	192.168.1.40	192.168.1.40		7A:80:35:54:E4:4D	
>	LKUSR-0974.logitek.local	192.168.1.41	Gemtek Technology Co., Ltd.	80:02:9C:CE:26:85	
>	192.168.1.201	192.168.1.201	Advanced Digital Broadcast SA	68:53:59:88:46:D2	

Figura 4.5: Rastreo de IPs en la red mediante Advanced IP Scanner

- Enroscada directamente a un tanque, gracias a su hardware de montaje con un diámetro exterior de 59mm.
- Usando su soporte para el montaje en superficies planas.
- Usando su soporte tipo U para su montaje en tubos.

En los dos últimos casos deben utilizarse los agujeros de tornillo situados a ambos lados de la caja [5].

4.2. Configuración en ThingPark Enterprise

4.2.1. Configuración del Gateway

Habitualmente y también en este caso, el Gateway ODU Macro V1.5 - 8 channels es vendido por Actility con su software LRR (Long Range Relay) ya instalado. La forma de acceder a su interfaz es vía SSH. SSH es un protocolo cuya principal función es el acceso remoto a un servidor mediante un canal seguro y cifrado.

Antes que nada, se necesita saber la IP que el router de red local ha asignado al Gateway LoRaWAN mediante DHCP (Dynamic Host Configuration Protocol). Como la puerta de enlace ya forma parte de nuestra red, su IP puede identificarse de varias maneras. Algunos ejemplos son:

- Accediendo a la web de administración del router de red local.
- Rastreado los dispositivos en nuestra red mediante softwares como Advanced IP Scanner. [Figura 4.5]

Utilizando este último, puede identificarse el dispositivo por su fabricante (Gemtek) así como por su dirección MAC.

Obtenida la IP, en este proyecto se ha utilizado el software Putty para conectar mediante SSH con el Gateway. En su ventana de inicio, hay que introducir la IP y determinar el puerto por el que se va a establecer comunicación (22 por defecto en SSH). [Figura 4.6]

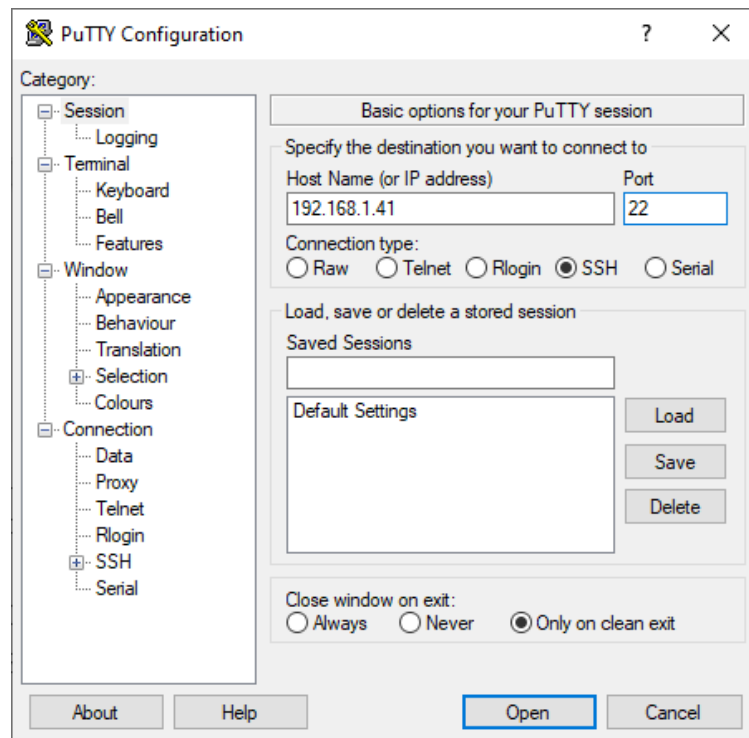


Figura 4.6: Conexión SSH mediante Putty

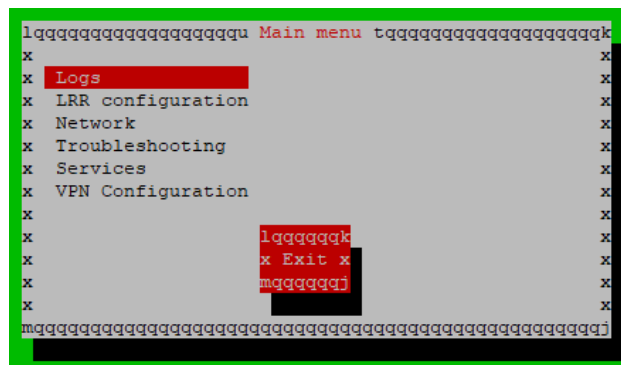


Figura 4.7: Menú principal del software LRR del Gateway

Al conectar, el equipo nos requiere unas credenciales para acceder al software LRR. El usuario a introducir siempre es 'support'. En cambio la contraseña es dada por Actility para cada dispositivo. Una vez introducidas las credenciales correctas se accede al menú principal de la interfaz del software LRR. [Figura 4.7]

Cada uno de estos menús contienen parámetros de configuración, funcionalidades o información del Gateway. Para poder dar de alta posteriormente el dispositivo en ThingPark Enterprise, se requiere seguir los pasos descritos a continuación:

- Obtener el LRR UID: el LRR UID es un código único que utiliza ThingPark Enterprise para identificar la instancia específica del software LRR que está corriendo en un Gateway. Para obtenerlo se debe acceder a los menús LRR configuration >Get LRR UID. En la ventana resultante, se muestra el LRR

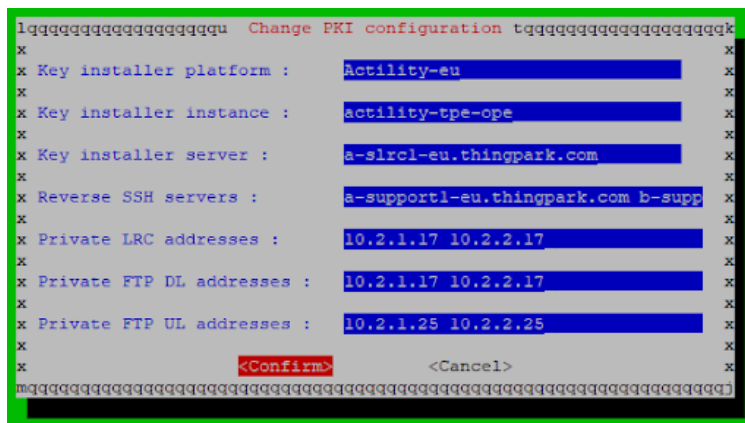


Figura 4.11: Configuración de los parámetros PKI del gateway

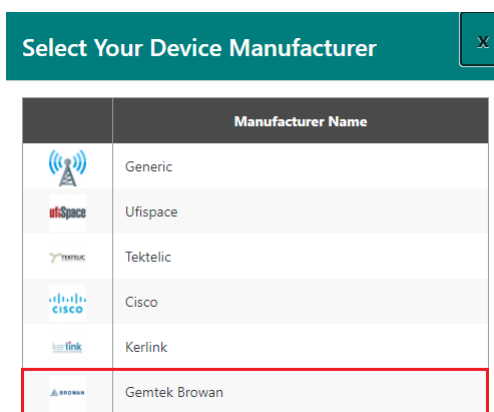


Figura 4.12: Creación de un nuevo gateway en ThingPark Enterprise

4.2.2. Dada de alta del Gateway en ThingPark Enterprise

Una vez establecida la configuración y recogidas sus claves, este puede ser dado de alta en el servidor red Thingpark enterprise. Para ello, se debe acceder al menú Base Stations >Create, disponible en la parte izquierda de la página principal. En la ventana resultante se escoge el fabricante del gateway con el que se desea comunicar, en este caso Gemtek Browan. [Figura 4.12]

En la siguiente ventana [Figura 4.13], se han de introducir los datos recopilados anteriormente, así como seleccionar el modelo y asignarle el nombre que tendrá en el Thingpark del usuario.

Rellenados y aplicados estos campos, el dispositivo constará como conectado a la plataforma. Durante los próximos dos minutos el equipo descargará los certificados y validará la conexión con el servidor red. Tras esto, el dispositivo queda totalmente listo para su uso, y así se indica en la plataforma. [Figura 4.14]

4.2.3. Creación del conector ThingPark Enterprise - Broker MQTT de Mosquitto

Previamente a dar de alta los nodos en Thingpark Enterprise, es necesario crear el conector que establecerá la comunicación entre la plataforma y el destino de los datos. Para ello, se debe acceder al menú

Enter Your Base Station Information*

Model [?]

ODU Macro v1.5 8-channels

Download the base station documentation

Download the base station image

Name [?]

Outdoor Gateway ✓

LRR UUID [?]

1C497B-80029CCE2685 ✓

RF Region [?]

EU 863-870MHz (8 channels)

Additional Information [?]

Write here...

Disable public key authentication

Public key [?]

```
-----BEGIN PUBLIC KEY-----
MIGfMADGCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCivQzaRIRtdcwVcsCC3BKIGjft
aZiFLKGBdJT4qNLRDo1MpBIR0GArtjOtrUqX1mKqBHUHytxbXQXQkGfy2AiQXns
kbVY140IAjvWSCaTvjh9VDZfXPRxVpNnsIb/s2P46svLJse/isiBEIE3hFM/N9D4
IDvEzTIMv3wTPK325uIDACAR
```

Figura 4.13: Establecimiento de parámetros del gateway en ThingPark Enterprise

BASE STATION STATUS

Connection: ● ACTIVE

Network Connection: 📶 CONNECTED 🚫 NOT ACTIVE

LoRaWAN™ Radio Status: 📡 STARTED

Clock Synchronization: 🕒 NTP SYNCHRONIZED

Base Station Restart Time
Today - 16:58:19

Last Uplink [?]
Today - 18:07:40

Last Downlink [?]
Today - 18:07:46

Figura 4.14: Indicadores de estado del gateway en ThingPark Enterprise

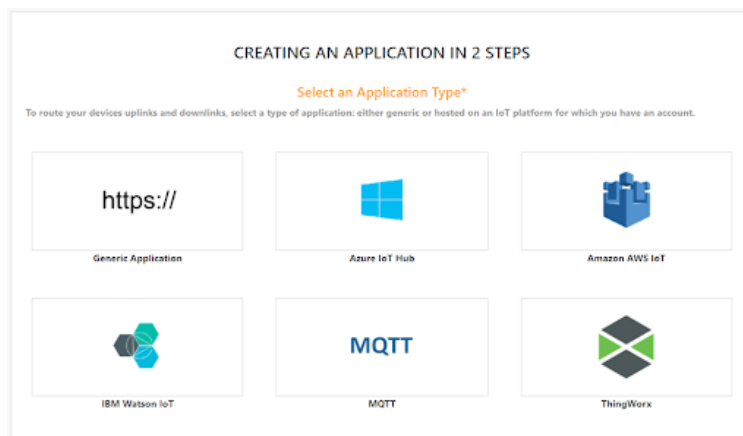


Figura 4.15: Tipos de salidas de datos en ThingPark Enterprise

Set Your Application*

Enter the values corresponding to your MQTT server parameters. Retrieve the values?

Name [?]

Mosquito MQTT Server ✓

Content Type [?]

JSON enriched document

Downlink Port [?]

1

Hostname [?]

test.mosquitto.org:1883 ✓

Account prefix [?]

ActivityGW ✓

Protocol [?]

SSL

Figura 4.16: Parámetros del conector MQTT en ThingPark Enterprise

izquierdo Applications >Create. Aparece ventana de la figura 4.13, en la que se listan los diferentes tipos de aplicaciones de salida.

En este caso, se selecciona la opción MQTT para establecer comunicación con el servidor MQTT de Mosquitto. En la siguiente ventana se introduce la dirección del servidor MQTT junto al puerto mediante el que se comunicará. Como este es un proyecto de estudio y no se ha configurado seguridad, se selecciona el puerto 1883, puerto por defecto del protocolo. [Figura 4.16]

Cabe destacar el campo 'Account prefix', que será el Topic bajo el que se publicarán los datos de todos los dispositivos que se asocien a esta aplicación.

Enter Your Device Information*

Model*	LoRaWAN 1.1 revA - class A	
Name*	Sensor TH	✓
DevEUI*	70-B3-D5-9B-A0-00-9F-7D	✓
Activation mode*	Over-the-Air Activation (OTAA)	
JoinEUI*	70-B3-D5-9B-A0-00-00-04	✓
NwkKey*	D2-1A-1A-67-12-DA-D1-B8-CC-4F-6D-D1-4E-9A-8E-2F	✓
AppKey*	CF-57-60-3F-4C-CB-C4-89-63-A9-6E-44-2B-D7-58-8A	✓

Figura 4.17: Creación de un dispositivo LoRa genérico en ThingPark Enterprise

4.2.4. Dada de alta de los sensores en ThingPark Enterprise

El siguiente paso es dar de alta los nodos en el servidor red de Actility. Esto se realiza desde el menú izquierdo Devices >Create. De igual manera que a la hora de crear el Gateway, se debe seleccionar el fabricante de cada dispositivo. Como Atim no consta entre estos, se selecciona la opción genérica para dispositivos LoRaWAN.

En la ventana resultante, se deben rellenar los siguientes campos:

- Modelo: versión de LoRaWAN que soporta el nodo y su clase (A, B o C).
- Nombre: nombre que tendrá el dispositivo en la plataforma Thingpark.
- Claves del nodo: DevEUI, AppEUI, NwkKey y AppKey.
- Aplicación: asociación con el conector MQTT recientemente creado.

Una vez establecidas estas configuraciones, el servidor red Thingpark Enterprise está publicando los datos de los nodos configurados en el servidor MQTT de Mosquitto.

4.3. Acondicionamiento con Node-RED

Como ya se ha visto en anteriores dispositivos, el payload que transmiten los nodos no indican el valor crudo medido. En la mayoría de los casos, el payload está conformado por una cadena de caracteres hexadecimales, algunos de los cuales se han de separar y acondicionar para obtener la lectura real. Para realizar esta labor se ha escogido la utilidad Node-RED.

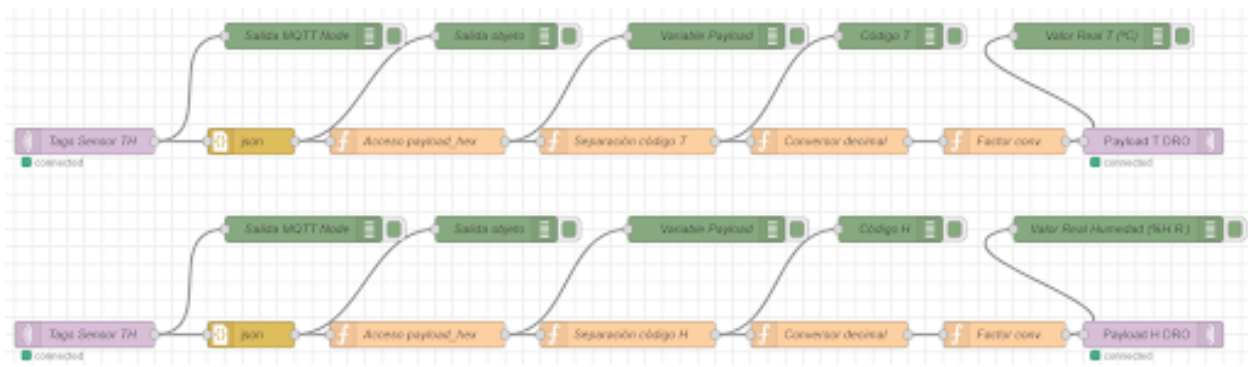


Figura 4.18: Flujos de acondicionamiento del sensor TH

Node-RED es una herramienta de desarrollo basada en programación visual de flujos para conectar dispositivos, APIs y servicios online, principalmente orientada a aplicaciones IoT. Esta proporciona un editor de flujos accesible a través de cualquier navegador de Internet, el cual permite crear funciones en lenguaje de programación JavaScript. Cada una de estas funciones está visualmente representada por un bloque (nodo). Node-RED ofrece nodos predeterminados con funcionalidades comunes, así como la posibilidad de crear nodos con funciones propias.

4.3.1. Flujo para el sensor de temperatura y humedad

Flujo para el sensor de temperatura y humedad El payload enviado por el dispositivo ACW/LW8-TH consta de 10 caracteres hexadecimales. Tal y como se ha citado en la sección referente a este, los caracteres del 3º al 6º corresponden al código de temperatura, el cual se debe convertir a decimal y aplicar un factor de conversión para obtener la temperatura medida. De igual manera, se debe seguir este proceso con los caracteres del séptimo al décimo para obtener el valor crudo de porcentaje de humedad medido.

En la figura 4.18 se muestran los flujos para separar y acondicionar ambos códigos:

Los nodos verdes son simples funciones para visualizar el mensaje (en el mismo editor) en cada punto deseado del flujo.

De izquierda a derecha, se citan los diversas modificaciones por las que pasan ambos payloads:

- Primeramente, el nodo de entrada MQTT (color lila) permite a la herramienta comunicarse con un servidor MQTT y leer variables de entrada de un determinado topic. Este conector lee el string en formato JSON (JavaScript Object Notation) que contiene todas las variables enviadas por Thingpark Enterprise, entre ellas el payload del sensor.
- Para poder acceder a cada una de las variables de forma independiente, es necesario convertir el mensaje JSON a objeto JavaScript. Esta función es realizada por el nodo 'json' (amarillo). En el siguiente código se muestra el formato del string JSON de entrada al flujo. En este ejemplo, 'DevEUI_uplink'

es la cabecera del mensaje, el cual contiene el valor de 4 variables:

```
{
  "DevEUI_uplink": {
    "Time": "2021-08-13T16:06:38.49+02:00",
    "DevEUI": "70B3D59BA0009F7D",
    "FCntUp": "23",
    "payload_hex": "176b948d96"
  }
}
```

El formato del objeto JavaScript DevEUI_uplink resultante de la conversión es:

```
DevEUI_uplink {
  Time: "2021-08-13T16:06:38.49+02:00",
  DevEUI: "70B3D59BA0009F7D",
  FCntUp: "23",
  payload_hex: "176b948d96"
}
```

- Los siguientes nodos (naranjas) estan creados con código propio. En cada uno de estos se crea una nueva variable 'message' acondicionando el mensaje proveniente del nodo anterior.

El primero de estos, separa la variable payload_hex (cadena de caracteres hexadecimales) del resto de las variables presentes en el objeto JavaScript mediante la línea de código:

```
var newMsg = { payload: msg.payload.DevEUI_uplink.payload_hex };
return newMsg;
```

Tras esto, se separan los caracteres correspondientes al código de temperatura y humedad según el flujo:

```
var newMsg = { payload: msg.payload[2] + msg.payload[3] + msg.payload[4] + msg.p
return newMsg;
```

Estos códigos se convierten a formato decimal mediante la función parseInt (.), indicando que el valor de entrada está en formato hexadecimal (base 16):

```
var newMsg = { payload: parseInt(msg.payload,16) }; return newMsg;
```

Finalmente, se aplica el factor de conversión indicado en el manual del sensor para obtener el valor real leído. Por ejemplo para el caso de la temperatura:

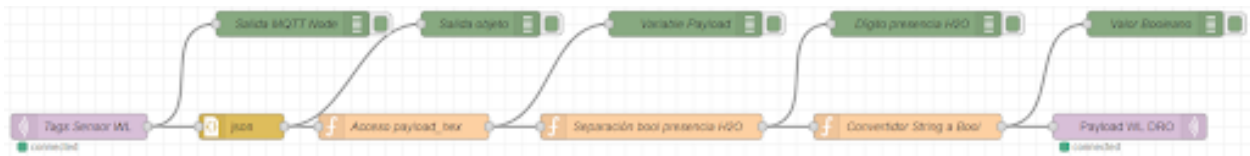


Figura 4.19: Flujo de acondicionamiento del sensor de fuga de agua

```
var newMsg = { payload: ((msg.payload * 175.72)/65536) - 46.85 };
return newMsg;
```

Acondicionados los datos, se usa un nodo de salida MQTT para publicar los valores en un topic diferente del servidor MQTT de Mosquitto.

4.3.2. Flujo para el sensor de fuga de agua

El payload enviado por el ACW/LW8-WL está formado por 6 caracteres. Cuando sus 2 terminales que realizan lecturas detectan un estado diferente, el dispositivo envía hasta 5 payloads (espaciados en intervalos de 10 minutos) con este valor. La presencia de agua es un dato de tipo booleano que viene indicado por el último carácter del payload.

En la figura 4.19 se muestra el flujo para obtener el dato.

Como en los casos anteriores, se separa los caracteres de interés del resto de datos presentes en el string JSON. Finalmente, se ha aplicado un conversor para modificar el tipo de dato de string a bool, cuyo código es el siguiente:

```
if (msg.payload === '1'){
    msg.payload=true;
}
if (msg.payload === '0'){
    msg.payload=false;
}
return msg; d
```

4.3.3. Flujo para el sensor de nivel

El payload enviado por el dispositivo ACW/LW8-LVL consta de 8 caracteres hexadecimales, cuyos 4 últimos corresponden al código hexadecimal de distancia en centímetros.

En la figura 4.20 se muestra el flujo para obtener el dato real leído.

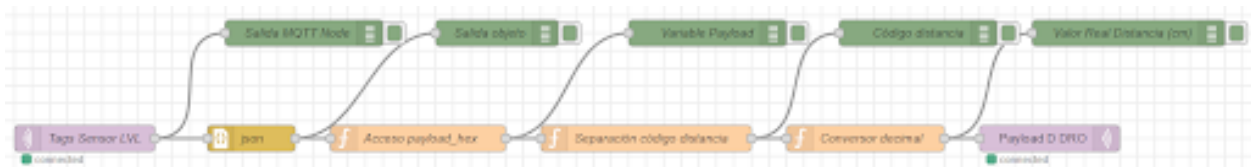


Figura 4.20: Flujo de acondicionamiento del sensor de nivel

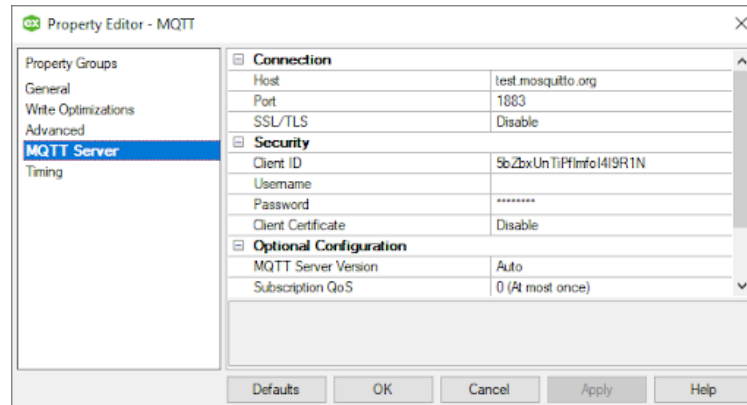


Figura 4.21: Configuración del canal MQTT en KEPServerEX

A diferencia del sensor de temperatura y humedad, el código de este corresponde al valor real de distancia, por lo que no se requiere aplicar un factor de conversión. En este caso, se han de seguir los mismos pasos para separar el código y finalmente convertirlo a formato decimal para obtener el valor crudo.

4.4. Creación del proyecto de KEPServerEX

En este apartado se citan el procedimiento seguido a la hora de crear el proyecto de KEPServerEX, cuya principal función es hacer accesibles mediante OPC UA los datos obtenidos a través de MQTT. Además, también se ha aplicado la configuración necesaria para hacer posible el envío de downlinks desde el cliente final SCADA.

4.4.1. Suscripción a los topics del Broker

El primer paso consiste en la obtención de los datos, tanto los publicados desde Thingpark Enterprise como desde Node-RED, disponibles en el servidor MQTT de Mosquitto. Para ello, se crea un canal de comunicación, indicando que este va a utilizar el driver MQTT de KEPServerEX. En la configuración de este cliente MQTT se establecen los parámetros básicos para conectar con el broker de Mosquitto, como su dirección (host name) y el puerto para comunicar. [Figura 4.21]

De igual manera, en esta ventana se indica que la conexión se va a establecer sin seguridad, por lo que no es necesario asignar ningún usuario. El identificador del cliente es generado de forma automática por el driver.

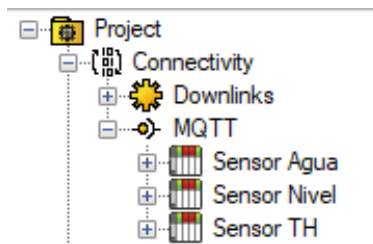


Figura 4.22: Canales y dispositivos creados en el proyecto de KEPServerEX

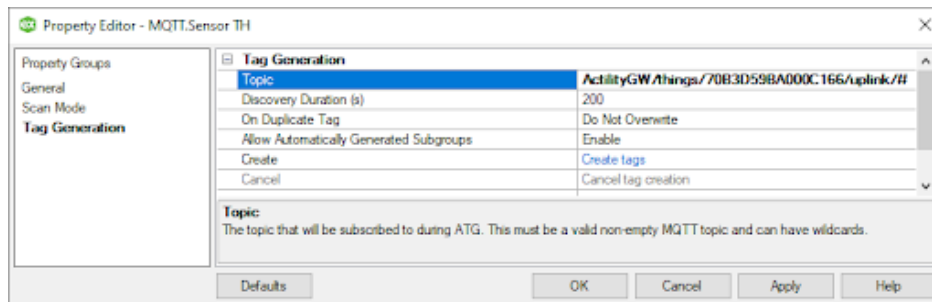


Figura 4.23: Acceso al topic del sensor TH desde KEPServerEX

Con tal de acceder a nuestros Topics presentes en el broker de Mosquitto, es necesario añadir un dispositivo en el canal creado. Aunque no es necesario, se crea un dispositivo para cada uno de los nodos. [Figura 4.22]

En las propiedades de cada uno de estos, se accede a la sección ‘Generación de Tags’ para acceder a los Topics de nuestra red LoRaWAN. Por ejemplo para obtener los tags del sensor de temperatura y humedad, se configuran los parámetros tal y como se muestra en la figura 4.23.

Siendo /things/<DevEUI>/uplink los subtopics que por defecto genera ThingPark Enterprise a la hora de publicar los datos de cada dispositivo. Una vez hecho esto, se hace clic en la opción ‘Create tags’ y se espera el tiempo determinado de búsqueda.

Un detalle a tener en cuenta es que el driver solo es capaz de generar estos tags si su valor se ha actualizado durante el período de búsqueda. Una sencilla solución para esto es establecer el tiempo de búsqueda a 600s (10 minutos), que es el máximo intervalo de tiempo entre 2 uplinks enviados por los nodos.

En la parte izquierda de la figura 4.24 se muestran los Topics a los que cada dispositivo está suscrito. Adicionalmente, en la ventana central se muestran los tags generados para el Topic del sensor de temperatura y humedad mencionado en este mismo apartado.

Los tags bajo los Topics ‘ActilityGW’ corresponden a todas las variables directamente publicadas desde ThingPark Enterprise las cuales no han sido acondicionadas. En cambio, los tags bajo los Topics introducidos por ‘Payload’ corresponden a los payloads acondicionados mediante la aplicación Node-RED.

Tag Name	Address	Data Type	Scan Rate
Feedback	AcctlyGW.Frange/08203056A0009F7D1UpLink#Feedback	String	100
DevE13_LupLink_ADFBst	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_ADFBst	Lamp	100
DevE13_LupLink_Channel	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Channel	String	100
DevE13_LupLink_CustomerDataAbNvr	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_CustomerDataAbNvr	String	100
DevE13_LupLink_CustomerDataAbNvr	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_CustomerDataAbNvr	String	100
DevE13_LupLink_CustomerID	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_CustomerID	String	100
DevE13_LupLink_DeVAddr	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_DeVAddr	String	100
Feedback_WL_DFD	AcctlyGW.Frange/08203056A0009F7D1UpLink#Feedback_WL_DFD	String	100
DevE13_LupLink_DeVtGr	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_DeVtGr	Lamp	100
DevE13_LupLink_DynamicsClass	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_DynamicsClass	String	100
DevE13_LupLink_FCstDn	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_FCstDn	Lamp	100
DevE13_LupLink_FTCHlp	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_FTCHlp	Lamp	100
DevE13_LupLink_FTPrst	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_FTPrst	Lamp	100
DevE13_LupLink_Frequency	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Frequency	Double	100
DevE13_LupLink_InstandFFER	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_InstandFFER	Double	100
DevE13_LupLink_Late	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Late	Lamp	100
DevE13_LupLink_Load	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Load	String	100
DevE13_LupLink_LutESP	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutESP	Double	100
DevE13_LupLink_Lut	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Lut	String	100
DevE13_LupLink_LutLAT	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutLAT	Double	100
DevE13_LupLink_LutLDR	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutLDR	Double	100
DevE13_LupLink_LutRSZ	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutRSZ	Double	100
DevE13_LupLink_LutRSZ_Chain	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutRSZ_Chain	Lamp	100
DevE13_LupLink_LutRSZ_LutESP	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutRSZ_LutESP	Double	100
DevE13_LupLink_LutRSZ_Lut	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutRSZ_Lut	String	100
DevE13_LupLink_LutRSZ_LutRSZ	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutRSZ_LutRSZ	Double	100
DevE13_LupLink_LutRSZ_LutSFR	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutRSZ_LutSFR	Double	100
DevE13_LupLink_LutSNR	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_LutSNR	Double	100
DevE13_LupLink_MuarPFR	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_MuarPFR	Double	100
DevE13_LupLink_Muc_Just	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Muc_Just	String	100
DevE13_LupLink_Model3p	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Model3p	String	100
DevE13_LupLink_MType	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_MType	Lamp	100
DevE13_LupLink_Nb Trans	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Nb Trans	String	100
DevE13_LupLink_Instand_Just	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Instand_Just	Lamp	100
DevE13_LupLink_SpFact	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_SpFact	Lamp	100
DevE13_LupLink_SubBand	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_SubBand	String	100
DevE13_LupLink_Time	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_Time	String	100
DevE13_LupLink_TstFlow	AcctlyGW.Frange/08203056A0009F7D1UpLink#DevE13_LupLink_TstFlow	Double	100

Figura 4.24: Suscripción de cada dispositivo a los topics del broker

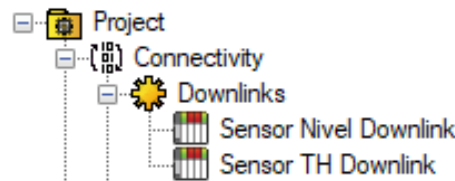


Figura 4.25: Dispositivos creados para el envío de downlinks

4.4.2. Creación de tags para el envío de downlinks

Para realizar el envío de mensajes descendentes a los nodos se utilizará el componente de KEPServe-rEX IoT Gateway. No obstante, primeramente es necesario crear tags en un canal de comunicación tipo Simulator, los cuales contendrán el payload del downlink.

Se crea el canal simulador con 2 dispositivos, uno para cada downlink [Figura 4.25]. Como ya se ha indicado en anteriores apartados, el sensor de presencia de agua ACW/LW8-WL no soporta la recepción de mensajes descendentes.

cmd

En cada dispositivo se crea un tag tipo String, ya que los downlinks deben estar conformados por cadenas de caracteres hexadecimales. Estos cuentan con la configuración mostrada en la figura 4.26.

Property Groups	Identification
General	Name: Sensor Nivel Downlink
Scaling	Description:
	Address: S0001
	Data Type: String
	Client Access: Read/Write
	Scan Rate (ms): 100
	Name: Specify the identity of this object.

Figura 4.26: Creación de los tags para el envío de downlinks

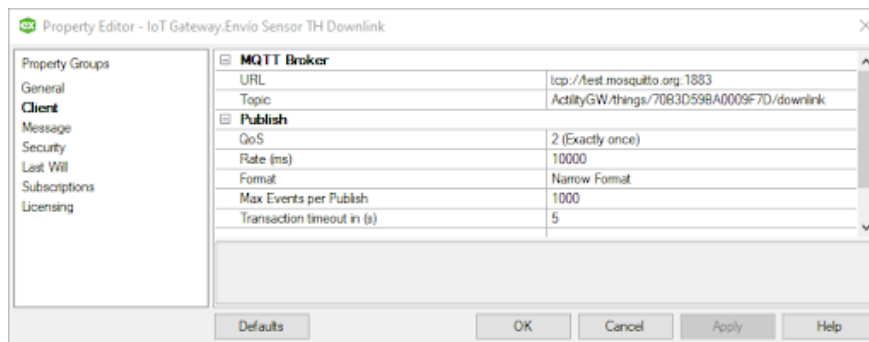


Figura 4.27: Propiedades del cliente MQTT publicador en KEPServerEX

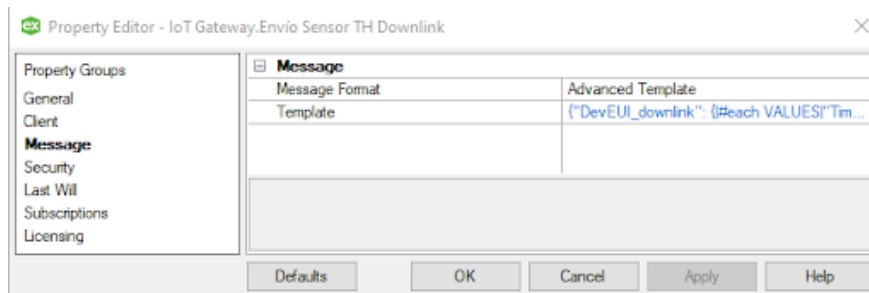


Figura 4.28: Selección de la plantilla avanzada de mensaje JSON

Las direcciones de KEPServerEX S0001, S0002... son registros internos del software que permiten al usuario asociarlos a un tag de los canales simulador. Así mismo, en el campo 'Client Access' se determina que los clientes conectados tengan permisos de lectura y escritura sobre este tag.

4.4.3. Creación del cliente MQTT publicador

El IoT Gateway es un plug-in opcional de KEPServerEX que permite al usuario publicar los tags de sistema y dispositivos en endpoints de terceros, utilizando protocolos estándar basados en IP como MQTT y REST.

Para publicar los downlinks de los sensores de temperatura/humedad y nivel se crean dos nuevos clientes MQTT. En las propiedades de estos se definen los siguientes parámetros. En la figura 4.27 se muestran los parámetros definidos para la sección 'Client'.

Se indica la dirección del servidor donde se pretende publicar los datos, así como el puerto para establecer la comunicación. Por otro lado se determina el topic en el que publicar los datos. Por defecto, ThingPark Enterprise espera encontrar los downlinks en el path: <Topic definido en TPE>/things/<DevEUI>/downlink.

Por otro lado, en el apartado 'Message' se debe seleccionar la opción 'Advanced'. Esto permite al usuario definir el formato de los mensajes JSON que publicará el cliente. [Figura 4.28]

En la figura 4.29 se muestra el formato y los parámetros mínimos necesarios para que ThingPark Enterprise reconozca el mensaje publicado como un downlink.

```

{
  "DevEUI_downlink": {
    "Time": "2019-07-10T15:38:46.882+02:00",
    "DevEUI": "0018B2000000B20",
    "FPort": 1,
    "payload_hex": "9e1c4852512000220020e3831071"
  }
}

```

Figura 4.29: Formato de downlink compatible con ThingPark Enterprise [66]

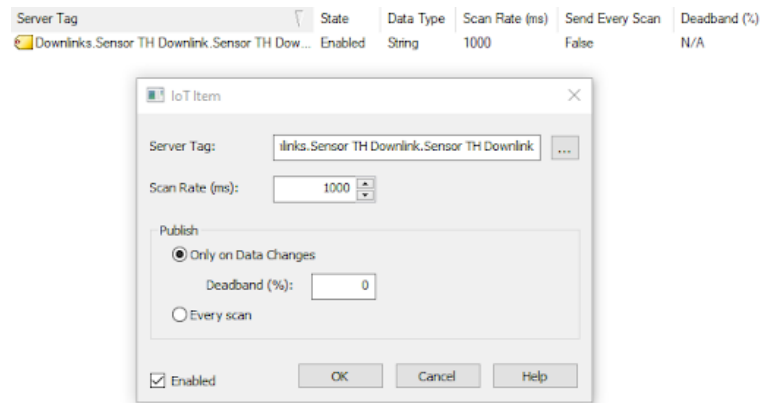


Figura 4.30: Ítem de asignación al tag a publicar

Por lo tanto, en KEPServerEX se define de igual manera, utilizando las variables internas de KEPServerEX (indicadas entre '|'):

```

{
  "DevEUI_downlink": {
    |# each VALUES|
      "Time": |SERVERTIMESTAMP| ,
      "DevEUI": "70B3D59BA0009F7D" ,
      "FPort": "1" ,
      "payload_hex": |VALUE|
    |/ each | }
}

```

La palabra clave 'each' es requerida para actualizar las variables internas |SERVERTIMESTAMP| y |VALUE| cada vez que publique un mensaje con esta plantilla [67]. Por último, en cada cliente debe crearse un ítem, que se asignará al tag cuyos valores se van a publicar. [Figura 4.30]

En esta ventana, se asigna el tag creado anteriormente en el canal simulador. Cabe recalcar que se ha de seleccionar la opción 'Only on Data Changes' para que el dato se publique una única vez cuando varíe su valor.

	Nombre	Tamaño de la matriz	Tipo
	Filtrar texto	Filtrar texto	(Todo)
1	Temperatura	0	Entero
2	Humedad	0	Entero
3	Presencia_Agua	0	Booleano
4	Sensor_TH_Downlink	0	Cadena
5	Sensor_Nivel_Downlink	0	Cadena
6	Distancia	0	Entero
7	Etiqueta_Sensor_TH	0	Cadena
8	Etiqueta_Sensor_Nivel	0	Cadena

Figura 4.31: Hoja de datos del proyecto en AVEVA Edge

Configuración del servidor

Nombre Conexión: KEPServerEX

Configuración

End point: opc.tcp://127.0.0.1:49320

Usuario:

Contraseña:

Tiempo de Esp.: 1000 ms

Seguridad...

Avanzado...

Probar Conexión

Aceptar

Cancelar

Figura 4.32: Propiedades de la conexión OPC UA con KEPServerEX

4.5. Desarrollo del SCADA en AVEVA Edge

En las próximas secciones se echa un vistazo general a las tareas y procedimientos realizados a la hora de desarrollar la aplicación SCADA del presente proyecto. No se pretende detallar punto por punto, ya que algunos procesos y programas son similares en varios elementos de la interfaz, únicamente variando las variables afectadas.

4.5.1. Creación de los tags de proyecto y comunicación con el servidor OPC

Los tags de proyecto son todas aquellas variables definidas en el proyecto de AVEVA Edge, tanto si estas se utilizan en la misma aplicación como si no. Para crearlos debe accederse al menú Global del Explorador de Proyectos y abrir la Hoja de datos haciendo doble clic. [Figura 4.31]

Esto desplegará la hoja de datos en la parte central de la pantalla. En ella, se insertan los tags deseados, definiendo su nombre, tipo de dato, y permisos de acceso a los clientes.

Una vez creados los tags en el proyecto de AVEVA Edge, se procede a habilitar su conexión con KEPServerEX. Para ello, se accede a la pestaña Comm del Explorador de Proyectos y se crea una nueva conexión OPC UA. En las propiedades de esta, se le da nombre y se define el endpoint de la interfaz UA de KEPServerEX, en este caso `opc.tcp://127.0.0.1:49320`. [Figura 4.32]

Se crea un nuevo cliente OPC UA y se le asigna la conexión recién creada. En la tabla que se muestra

Nombre de Tag	Browse Path	Escanear	Div	Añadir	ID del nodo:
1 Temperatura	0:Objects/2/MQTT/2/Sensor TH/2/Payload T DRO/2/Payload	Siempre			ns=2;s=MQTT.Sensor TH.Payload T DRO.Payload
2 Humedad	0:Objects/2/MQTT/2/Sensor TH/2/Payload H DRO/2/Payload	Siempre			ns=2;s=MQTT.Sensor TH.Payload H DRO.Payload
3 Presencia_Agua	0:Objects/2/MQTT/2/Sensor Agua/2/Payload WL DRO/2/Payload	Siempre			ns=2;s=MQTT.Sensor Agua.Payload WL DRO.Payload
4 Nivel_Contenedor	0:Objects/2/MQTT/2/Sensor Nivel/2/Payload D DRO/2/Payload	Siempre			ns=2;s=MQTT.Sensor Nivel.Payload D DRO.Payload
5 Sensor_Nivel_Downlink	0:Objects/2/Downlinks/2/Sensor Nivel Downlink/2/Sensor Nivel Downlink	Siempre			ns=2;s=Downlinks.Sensor Nivel Downlink.Sensor Nivel Downlink
6 Sensor_TH_Downlink	0:Objects/2/Downlinks/2/Sensor TH Downlink/2/Sensor TH Downlink	Siempre			ns=2;s=Downlinks.Sensor TH Downlink.Sensor TH Downlink
7 RSSI_TH	0:Objects/2/MQTT/2/Sensor TH/2/ActilityGW/2/things/2:70B3D59BA0009F7D/2/uplink/2:DevEUI_uplink&LnRSSI	Siempre			ns=2;s=MQTT.Sensor TH.ActilityGW/things.70B3D59BA0009F7D.uplink.DevEUI_uplinkLnRSSI
8 RSSI_WL	0:Objects/2/MQTT/2/Sensor Agua/2/ActilityGW/2/things/2:70B3D59BA000890C/2/uplink/2:DevEUI_uplink&LnRSSI	Siempre			ns=2;s=MQTT.Sensor Agua.ActilityGW/things.70B3D59BA000890C.uplink.DevEUI_uplinkLnRSSI

Figura 4.33: Conexión de tags entre AVEVA Edge y KEPServerEX

en la figura 4.33, se define por un lado el nombre de los tags en AVEVA Edge y por otro el path del tag equivalente en KEPServerEX. Esto es tarea sencilla gracias al explorador OPC integrado en el software.

Para este proyecto, se ha conectado con los tags de KEPServerEX correspondientes a:

- Valores leídos por los sensores.
- Downlink a enviar a los nodos.
- Calidad de señal de los mensajes.

4.5.2. Creación y diseño de pantallas

El primer paso para empezar a desarrollar la aplicación es determinar qué pantalla será la de inicio. Esto puede hacerse en la configuración del proyecto. En la cinta superior, se accede a la pestaña Proyecto, dentro del apartado configuración se selecciona la opción visor. Aparece la ventana de la figura 4.34.

En ella, el usuario puede modificar diversas opciones referentes a la visualización del proyecto. En el campo Pantalla Inicio, se define nuestra pantalla principal, en este caso main.scc. Esta pantalla puede crearse posteriormente.

Hecho esto, se crean las tres pantallas que conforman este proyecto: main, vista_planta y vista_historico. Esto debe hacerse desde la pestaña Gráficos del Explorador de Proyectos, haciendo clic derecho en el directorio pantallas y seleccionando la opción Insertar.

A continuación, se repasa la lógica y elementos gráficos encontrados en cada una de las pantallas.

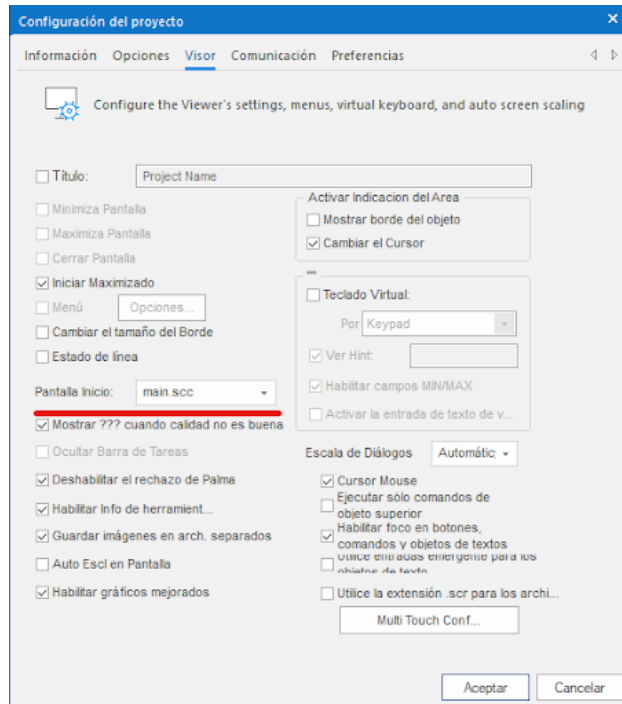


Figura 4.34: Configuración del proyecto de AVEVA Edge

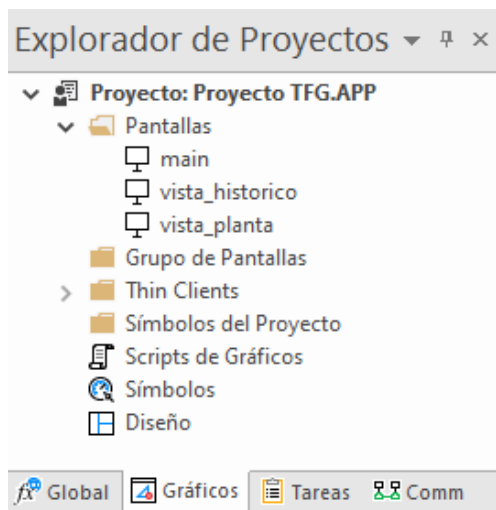


Figura 4.35: Pantallas del proyecto de AVEVA Edge



Figura 4.36: Interfaz de la pantalla principal del proyecto

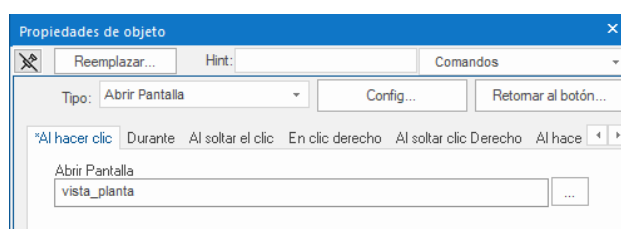


Figura 4.37: Propiedades del botón 'Vista de la planta'

4.5.2.1. Pantalla principal

La pantalla principal desarrollada en este proyecto presenta la interfaz mostrada en la figura 4.36.

Dejando de lado los elementos simplemente gráficos, como colores, formas y textos, en la parte central se muestran dos botones de acceso a las dos pantallas restantes. En estos, se ha aplicado un comando del tipo 'Abrir pantalla' al ser pulsados, definiendo la pantalla que debe abrirse respectivamente. En la figura 4.37 puede apreciarse la configuración para el caso del botón 'Vista de la planta'.

En la parte inferior izquierda de la pantalla, se muestra una salida de los valores analógicos medidos por los nodos. Estos son conformados por un cuadro de texto el cual es asociado con el valor del Tag de Proyecto correspondiente. En la figura 4.38 se muestra esta asignación para el caso de la salida de temperatura.

A su derecha, se ha diseñado un pequeño panel para mostrar las alarmas activas actualmente. Hay 4 alarmas posibles, cuya activación corresponde a rangos altos y bajos de temperatura y humedad. Para programar cuando se muestran estas alarmas, se les ha asignado al cuadro de texto una animación de visibilidad, cuyo disparador es un tag específico. En el siguiente caso, mostrado en la figura 4.39, para la alarma de temperatura se ha creado el tag V_Alarma_Tbaja.

Para programar la lógica de esta variable, se ha creado un nuevo script (en lenguaje VBScript) 'Cli-

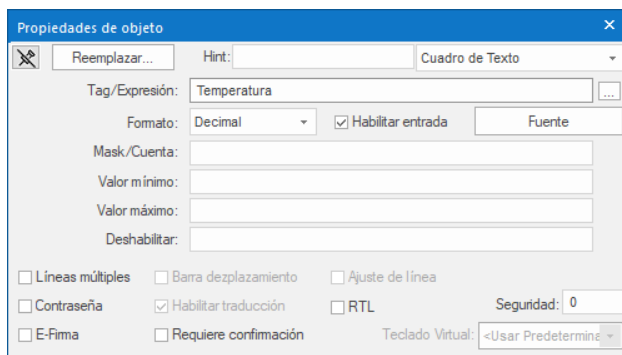


Figura 4.38: Asignación del cuadro de texto al valor de temperatura

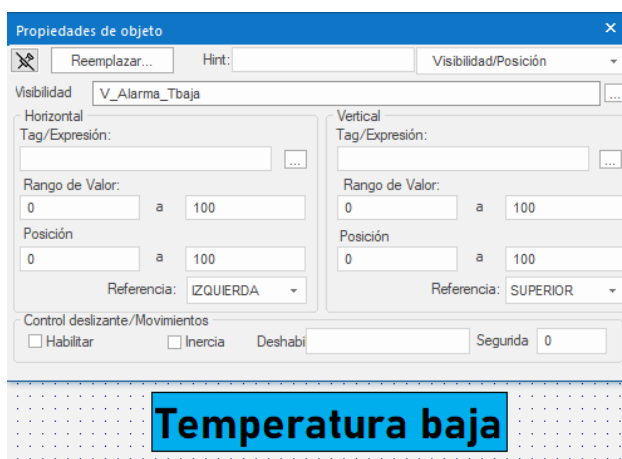


Figura 4.39: Animación de visibilidad para alarma de temperatura

matizador T', cuya parte del código también hace referencia a variables que serán vistas en la pantalla 'vista_planta'.

```
If $Temperatura > 25 Then
    Do Until $var3=500000
        $var3=$var3+1
        $Climatizador=1
        $V_Alarma_Talta=1
    Loop
    $V_Alarma_Talta=0
    $Climatizador=0
    $Temperatura = 25
End If

If $Temperatura < 17 Then
    Do Until $var4=500000
        $var4=$var4+1
        $Climatizador=1
        $V_Alarma_Tbaja=1
    Loop
    $V_Alarma_Tbaja=0
    $Climatizador=0
    $Temperatura = 17
End If

$var3=0
$var4=0
```

Las variables 'var' son simples contadores ascendentes para determinar el tiempo durante el que se ejecutan los bucles.

Por último en la parte superior derecha se ha posicionado un botón de cruz para salir de la aplicación utilizando la función interna del software \$Shutdown(), para que esta se ejecute al hacer clic sobre él. Adicionalmente, en la esquina inferior derecha se han añadido un par de cuadros de texto, cuya salida corresponde a los valores de los tags internos Date y Time, para mostrar la fecha y la hora en tiempo real.

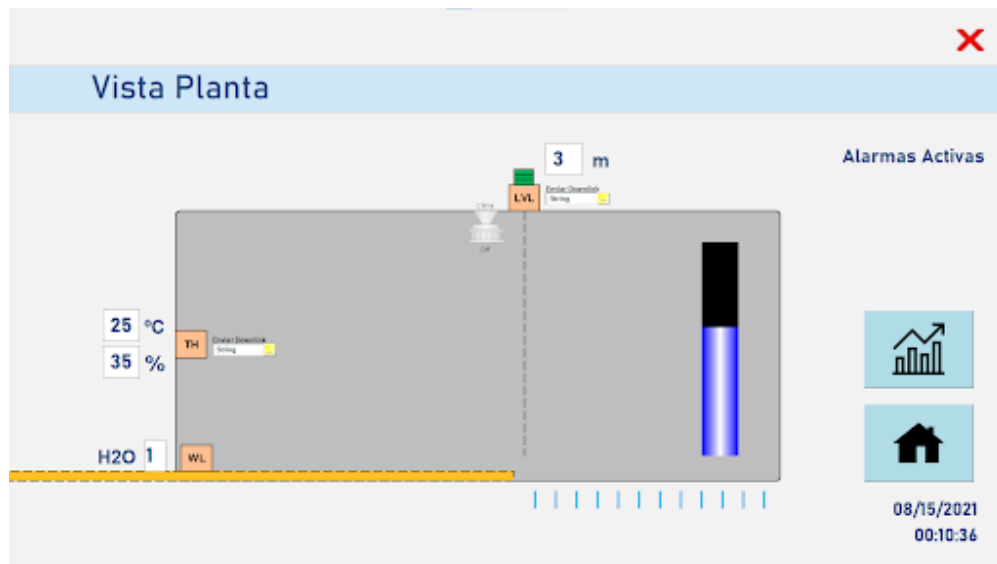


Figura 4.40: Interfaz de la pantalla 'Vista Planta'

4.5.2.2. Vista de la planta

La figura 4.40 muestra la interfaz de la pantalla 'vista_planta', donde se muestra un sistema siendo simulado con las variables recibidas de los nodos.

En el diagrama se pretende representar un contenedor que almacena fruta recién lavada, cuyo sistema sigue las siguientes normas:

- En la parte superior se posiciona el sensor de nivel, cuyo valor medido se representa por el cuadro de texto a su derecha y por la barra color azul. Este también dispone de una entrada de texto para introducir los downlinks que se deseen mandar al nodo.
- En la parte izquierda del contenedor se encuentra el sensor de temperatura y humedad, los valores del cual se muestran en los cuadros de texto a su izquierda. Como en el caso anterior, también dispone de una entrada para el envío de downlinks.
- Cuando los valores leídos de temperatura y humedad desbordan los límites máximos y mínimos, se activa el climatizador localizado en la parte superior del contenedor. Pasados unos segundos, el valor de temperatura/humedad se estabiliza a su límite más cercano. Esta lógica se ha definido en el script anterior.
- En la esquina inferior derecha, se posiciona el sensor de presencia de agua. Cuando este detecta, se abre la compuerta inferior para liberar el agua acumulada en el fondo del contenedor y evitar que se estopee la fruta. Para determinar el desplazamiento de la compuerta se ha añadido una animación de posición a su figura, dependiente del valor detectado de presencia de agua. [Figura 4.41]

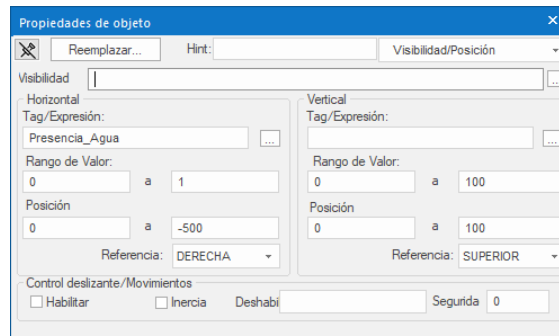


Figura 4.41: Animación de posición de la compuerta inferior

Así mismo se ha programado la visibilidad del agua cayendo cuando se abre la compuerta con el siguiente script:

```

If $Presencia_Agua = 1 Then
    Do Until $Presencia_Agua = 0
        $agua_visible_1=1
        Do Until $var1=40000
            $var1=$var1+1
        Loop
        $var1=0
        $agua_visible_1=0
        $agua_visible_2=1
        Do Until $var2=40000
            $var2=$var2+1
        Loop
        $var2=0
        $agua_visible_2=0
    Loop
End If

```

Encima de cada nodo, se ha posicionado un indicador, cuyas barras verdes activas indican la calidad de señal (RSSI) del último mensaje recibido del nodo. Esta lógica se ha programado con este script para el sensor de temperatura y humedad:

```

If $RSSI_TH < 0 And $RSSI_TH >= -50 Then
    $barra_señal_TH_1=1
    $barra_señal_TH_2=1

```



Figura 4.42: Interfaz de la pantalla 'Histórico de valores'

```

$barra_señal_TH_3=1
ElseIf $RSSI_TH < -50 And $RSSI_TH >= -65 Then
    $barra_señal_TH_1=1
    $barra_señal_TH_2=1
    $barra_señal_TH_3=0
ElseIf $RSSI_TH < -65 And $RSSI_TH >= -80 Then
    $barra_señal_TH_1=1
    $barra_señal_TH_2=0
    $barra_señal_TH_3=0
Else
    $barra_señal_TH_1=0
    $barra_señal_TH_2=0
    $barra_señal_TH_3=0
End If

```

En la parte derecha de la pantalla, se han posicionado el panel de alarmas activas y los botones de acceso al resto de pantallas.

4.5.2.3. Vista del histórico de valores

En la figura 4.42 se muestra la interfaz diseñada para la pantalla 'historico_valores'.

En esta, se muestran tres gráficos con los valores que han obtenido las variables durante la última hora.

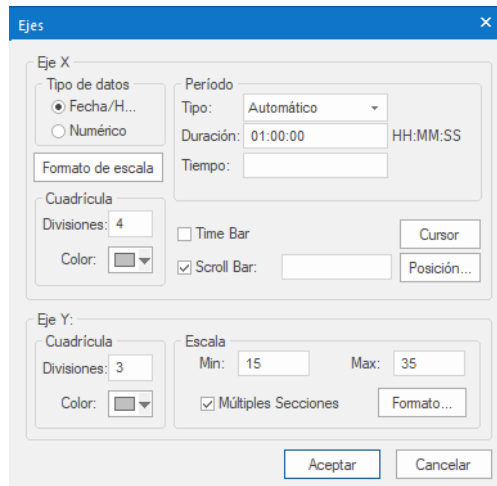


Figura 4.43: Configuración de los ejes del histórico

Con la barra desplazable inferior en cada uno de estos, pueden visualizarse valores anteriores. Se ha descartado historizar la variable Nivel ya que no tiene impacto directo en ninguna de las salidas del sistema (climatizador, compuerta).

El primer paso para crear estos gráficos ha sido determinar los valores de qué tags se van a almacenar en archivos internos del software. Para ello, en la ventana Tareas del Explorador de archivos se ha insertado una nueva Tendencia. En esta, se han seleccionado los tags Temperatura, Humedad y Presencia_Agua.

Una vez registrados los valores, se inserta un objeto de datos tipo Tendencia en la pantalla. Para el gráfico de la variable Humedad, primeramente se han modificado visualmente las líneas y textos. Posteriormente, se ha aplicado la configuración de los ejes, para hacer los datos entendibles al usuario. [Figura 4.43]

Capítulo 5

Resultados de la plataforma

La presente sección tiene como principal objetivo destacar los resultados más tangibles respecto al funcionamiento de la plataforma implementada.

Estos resultados pueden comenzar a verse en tiempo real en la aplicación Node-RED. En esta, accediendo al menú derecho Debug pueden observarse las salidas de los bloques tipo debug definidos en los diferentes flujos. Estos bloques muestran en pantalla el contenido que está pasando el punto del flujo en que están situados. En el caso de esta aplicación, se ha colocado un nodo debug a la salida de cada nodo con código. [figura 5.1]

En la figura 5.2 se muestra la salida del nodo debug al finalizar el flujo de acondicionamiento para obtener el valor de temperatura del sensor ACW/LW8-TH.

Desde que el sensor envía el mensaje de lectura hasta que este llega a la aplicación pasan aproximadamente unos 1.5 segundos.

El siguiente punto en el que se puede observar la actualización de los valores de las variables es en el proyecto de KEPServerEX. Este software viene con un cliente OPC DA integrado (Quick Client), desde el cual pueden visualizarse los valores de las variables en tiempo real. En la imagen 5.3 se muestra un

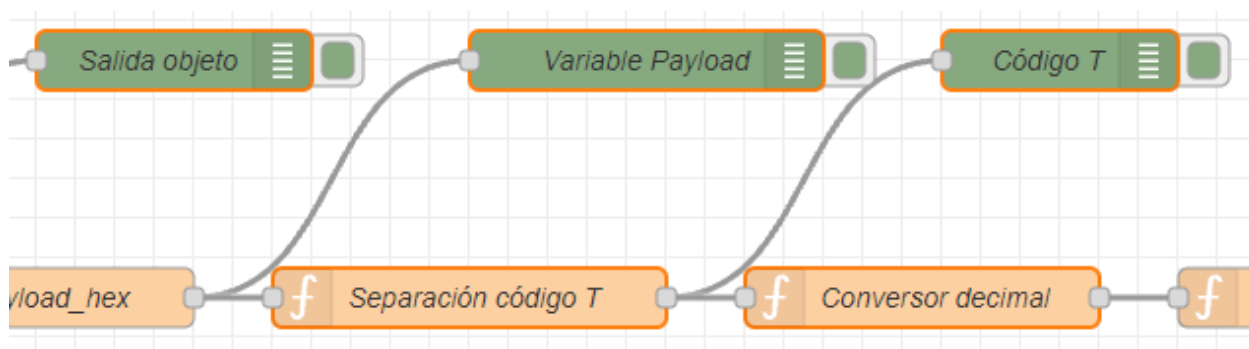


Figura 5.1: Nodos debug en los flujos

```
28/9/2021 22:13:50 node: Valor Real T (°C)
msg.payload : number
26.938671875000004
```

Figura 5.2: Salida del flujo para el acondicionamiento de la temperatura

MQTT.Sensor TH.ActilityGW.things.70B3D59BA0009F7D.uplink.DevEUI_uplink/payload_hex	String	176b5c967e	22:31:51.689	Good
MQTT.Sensor TH.ActilityGW.things.70B3D59BA0009F7D.uplink.DevEUI_uplink/SubBand	String	G2	22:31:51.689	Good
MQTT.Sensor TH.ActilityGW.things.70B3D59BA0009F7D.uplink.DevEUI_uplink/Time	String	2021-09-28T20:...	22:31:51.689	Good
MQTT.Sensor TH.ActilityGW.things.70B3D59BA0009F7D.uplink.Payload	String	{ "DevEUI_upl...	22:31:51.689	Good

Figura 5.3: Visualización de las variables en Quick Client

ejemplo de las variables que llegan directamente desde ThingPark Enterprise. Como puede observarse llega una variable nombrada Payload de tipo string, cuyo valor es el mensaje JSON que contiene cada una de las variables. Después, el driver MQTT de KEPServerEX separa cada una de las variables del mensaje en un tag distinto. Esto libera una gran carga en las comunicaciones del proyecto, ya que realmente se está leyendo solo un dato (la variable Payload) y no las 25 variables que contiene el mensaje completo. La última columna corresponde a la calidad de la lectura de cada variable. Esta puede obtener los valores Good, Bad o Uncertain.

Las comunicaciones entre Node-RED y KEPServerEX son casi inmediatas, de como máximo unos 0.5 segundos.

Por último en el SCADA desarrollado con AVEVA Edge, la actualización de los valores puede observarse en la ventana inferior Watch [figura 5.4], además de en la propia aplicación corriendo.

Este es el punto final de la red LoRaWAN desplegada. Las comunicaciones entre este y KEPServerEX vuelven a ser casi inmediatas. Esto implica que la tasa de actualización de los datos entre que un sensor envía un mensaje y este llega al SCADA es de aproximadamente 2.5 segundos.

Watch			
Tag/Expresión	Valor	Calidad	Continuar
Temperatura	26	BUENO	<input checked="" type="checkbox"/>
Humedad	35	BUENO	<input checked="" type="checkbox"/>
Distancia	122	BUENO	<input checked="" type="checkbox"/>

Figura 5.4: Visualización de las variables en el editor de AVEVA Edge 2020

Capítulo 6

Resumen del presupuesto

El cuadro 6.1 recoge el coste de los dispositivos y licencias necesarias para la implementación de la red LoRaWAN desarrollada en el presente proyecto.

Cuadro 6.1: Resumen del coste de la implementación de la red desarrollada

	Precio (€)
Nodo ACW/LW8-TH	165
Nodo ACW/LW8-LVL	369
Nodo ACW/LW8-WL	319
Gateway ODU Macro V1.5 - 8 channels	850
Licencia ThingPark Enterprise	1200
Licencias KEPServerEX	788
Licencia AVEVA Edge 2020	2740
Coste del personal	7200
Total	13631

Capítulo 7

Conclusiones

En el presente proyecto se ha desarrollado satisfactoriamente una red LoRaWAN privada. Esta red consta de 3 nodos LoRa, un Gateway, un servidor de terceros, una aplicación Node-RED, un servidor OPC y una aplicación SCADA. Es decir, se ha implementado una estructura completa de red y se ha aportado una solución para su integración en un entorno industrial real.

Para conocer el contexto de las tecnologías IoT y más en concreto la especificación LoRa, se ha estudiado su estado e implantación actual en diversas regiones. Pese a su gran potencial y avances en algunos países europeos, en España todavía está lejos de despuntar en el sector industrial. Algunos de los motivos que llevan a esta situación son la falta de conocimiento y la poca oferta de la tecnología LoRa en la península.

Bajo mi punto de vista, el futuro de las redes LoRaWAN es esperanzador. Cada vez las redes LPWAN están tomando más relevancia y eso hace que las plantas industriales las tengan en cuenta para satisfacer necesidades específicas. Su bajo coste de hardware, mantenimiento y implementación, y el bajo consumo energético las convierten en la mejor opción para hacer frente a las comunicaciones no críticas. Frente a sus competidoras, LoRaWAN resalta por su alto rango de alcance y la posibilidad de autogestión de la red. Esta ventaja respecto a su competencia queda reflejada en un estudio de la ABI research, el cual prevee que para el año 2026 LoRa sea la tecnología LPWA líder, utilizándose para más del 50% de las conexiones de estas redes [68].

Respecto a la comparación con otras tecnologías inalámbricas en auge como el 5G, se prevee que ambas coexistirán y se complementarán en el futuro. Por ejemplo en una situación que se tengan que recoger datos de dispositivos muy lejanos, LoRa sería la mejor opción para establecer la comunicación de una puerta de enlace con la gran masa de sensores, mientras que la red móvil 5G sería la conveniente para comunicar la puerta de enlace con el servidor de red. Esta forma de implementación ahorraría los costos del hardware, la energía y mantenimiento de sensores 5G.

Durante la realización del proyecto, se han encontrado inconvenientes que han implicado buscar y estudiar alternativas cuyas características encajaran con las necesidades planteadas. Nada más lejos de ser un problema, esto ha supuesto aprender y conocer mejor algunas herramientas como Node-RED, además de familiarizarse con lenguajes de programación como JavaScript y VBScript.

Gracias a la implantación de softwares comunmente utilizados en plantas, se ha demostrado la fácil integrabilidad de esta en la industria 4.0. Además de esto, el sistema es ampliamente escalable, pudiendo añadir e intercambiar nuevos nodos, Gateways e incluso clientes finales.

En este documento, quedan detalladamente redactados todos los pasos a seguir para implementar una red LoRaWAN desde cero con la plataforma ThingPark Enterprise. Además, se especifica el uso de esta en un ámbito industrial, comunicándola con softwares presentes en todas las plantas. Se abarca desde la inicialización y dada de alta de los nodos, hasta los procedimientos y configuración a seguir en los softwares de monitorización de datos. Esto aporta una información muy útil a cualquier planta industrial que esté planteando instalar una red de similares características.

Bibliografía

- [1] Oracle. ¿Qué es el Internet of Things (IoT). Consultado el 10 de marzo de 2021, <<https://www.oracle.com/es/internet-of-things/what-is-iot/>>
- [2] Logitek. Empezando a trabajar con LoRaWAN (I). Consultado el 10 de marzo de 2021, <<https://www.m2mlogitek.com/empezando-a-trabajar-con-lorawan-i/>>
- [3] Atim (23 de enero de 2018). Temperature & Hygrometry TH-I. Consultado el 18 de abril de 2021. Disponible en: <https://www.atim.com/en/produits/catalogue/acw_en/sensor-temperature-humidity-indoor-acw-th-i/>
- [4] Atim. Leakage detection sensor WL. Consultado el 19 de abril de 2021. Disponible en: <https://www.atim.com/en/produits/catalogue/acw_en/flood-detection-sensor-acw-wl/>
- [5] Atim. Ultrasonic distance measurement LVL. Consultado el 19 de abril de 2021, Disponible en: <https://www.atim.com/en/produits/catalogue/acw_en/level-measurement-acw-lvl/>
- [6] Browan (4 de septiembre de 2020). Outdoor Micro Gateway WAPS-232N Product Description. Consultado el 20 de abril de 2021. Disponible en: <<https://www.browan.com/product/outdoor-micro-gateway/detail>>
- [7] Actility. Actility Documentation: ThingPark Enterprise. Consultado el 22 de abril de 2021, <<https://docs.thingpark.com/thingpark-enterprise/6.1/Content/Home.htm>>
- [8] Aprendiendo Arduino. Qué es Node-RED. Consultado el 25 de abril de 2021, <<https://aprendiendoarduino.wordpress.com/2020/03/05/que-es-node-red/>>
- [9] Logitek. Un server OPC UA rápido y sencillo. Consultado el 26 de abril de 2021, <<https://www.kepsserverexopc.com/>>
- [10] AVEVA. AVEVA™ Edge: Herramienta de HMI e IoT Edge para OEM y usuarios finales . Consultado el 26 de abril de 2021, <<https://www.aveva.com/es-es/products/edge/>>

- [11] LoRaWAN. LoRaWAN, Descubre la mayor revolución para el Internet de las Cosas (IoT). Consultado el 10 de marzo de 2021, <<https://lorawan.es/>>
- [12] BMeters. New LoRaWAN Section. Consultado el 10 de marzo de 2021, <<https://www.bmeters.com/es/new-lorawan-section/>>
- [13] LoRa Alliance. About LoRa Alliance. Consultado el 10 de marzo de 2021, <<https://loralliance.org/about-lora-alliance/>>
- [14] Semtech. LoRa and LoRaWAN. Consultado el 10 de marzo de 2021, <<https://loradevelopers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/>>
- [15] LoRa Alliance. Webinars & Videos. Consultado el 12 de marzo de 2021, <<https://loralliance.org/webinars-videos/>>
- [16] N. Sornin (Semtech), M. Luis (Semtech), T. Eirich (IBM), T. Kramp (IBM), 6 O.Hersent (Actility), LoRaWAN Specification. Consultado el 16 de marzo de 2021, <<https://www.rs-online.com/designspark/rel-assets/ds-assets/uploads/knowledge-items/application-notes-for-the-internet-of-things/LoRaWAN%20Specification%201R0.pdf>>
- [17] Catsensors. Tecnología LoRa y LoRaWAN. Consultado el 18 de marzo de 2021, <<https://www.catsensors.com/es/lorawan/tecnologia-lora-y-lorawan>>
- [18] Minkafab. Dispositivos LoRaWAN. Consultado el 22 de marzo de 2021, <<https://minkafab.com/tecnologia-lora-lorawan/dispositivos-para-lorawan/>>
- [19] LoRa Alliance. Private LoRaWAN Networks, Your LPWAN Network Your Way. Consultado el 22 de marzo de 2021, <<https://pages.services/pages.lora-alliance.org/lorawan-private-networks/?ts=1607123605468>>
- [20] LoRa. Lora. Consultado el 22 de marzo de 2021, <<https://lora.readthedocs.io/en/latest/>>
- [21] The Things Network. The Things Node. Consultado el 23 de marzo de 2021, <<https://www.thethingsnetwork.org/docs/devices/node/>>
- [22] Medium. Haciendo IoT con LoRa: Capitulo 2. Tipos y Clases de Nodos. Consultado el 23 de marzo de 2021, <<https://medium.com/beelan/haciendo-iot-con-lora-capitulo-2-tipos-y-clases-de-nodos-3856aba0e5be>>
- [23] The Things Network. Device Classes. Consultado el 23 de marzo de 2021, <<https://www.thethingsnetwork.org/docs/lorawan/classes/>>

- [24] The Things Network. Security. Consultado el 30 de marzo de 2021, <<https://www.thethingsnetwork.org/docs/lorawan/security/>>
- [25] The Things Industries. OTAA vs ABP. Consultado el 30 de marzo de 2021, <<https://www.thethingsindustries.com/docs/devices/abp-vs-otaa/>>
- [26] Techplayon. LoRa – Device Activation Call Flow (Join Procedure) using OTAA and ABP. Consultado el 30 de marzo de 2021, <<https://www.techplayon.com/lora-device-activation-call-flow-join-procedure-using-otaa-and-abp/>>
- [27] Aprendiendo Arduino. Arduino y LoRaWAN. Consultado el 02 de abril de 2021, <<https://www.aprendiendoarduino.com/tag/lora-gateway/#:~:text=En%20las%20redes%20LoRaWan%20un,%20servidor%20de%20red%20centralizado.>>>
- [28] The Things Network. The Things Gateway. Consultado el 02 de abril de 2021, <<https://www.thethingsnetwork.org/docs/gateways/gateway/>>
- [29] Semtech. LoRaWAN Gateways. Consultado el 02 de abril de 2021, <<https://loradevelopers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/>>
- [30] Semtech. Network Server. Consultado el 04 de abril de 2021, <<https://loradevelopers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/>>
- [31] Logitek. Empezando a trabajar con LoRaWAN (III). Consultado el 04 de abril de 2021, <<https://www.m2mlogitek.com/empezando-a-trabajar-con-lorawan-iii/>>
- [32] The Things Industries. Application Server. Consultado el 05 de abril de 2021, <<https://www.thethingsindustries.com/docs/reference/components/application-server/>>
- [33] Semtech. Join Server. Consultado el 05 de abril de 2021, <<https://loradevelopers.semtech.com/library/tech-papers-and-guides/lora-and-lorawan/>>
- [34] Deloitte. IoT - Internet of Things. Consultado el 05 de abril de 2021, <<https://www2.deloitte.com/es/es/pages/technology/articles/IoT-internet-of-things.html>>
- [35] i-Scoop. Meaning, benefits and value of IIoT (Industrial Internet of Things). Consultado el 06 de abril de 2021, <<https://www.i-scoop.eu/internet-of-things-iiot/industrial-internet-things-iiot-saving-costs-innovation/industrial-internet-things-iiot/>>
- [36] Logitek. LPWAN: Qué son y para qué se utilizan. Consultado el 06 de abril de 2021, <<https://www.m2mlogitek.com/lpwan-que-son-y-para-que-se-utilizan/>>

- [37] Sigfox España. Preguntas frecuentes: ¿Qué es Sigfox?. Consultado el 07 de abril de 2021, <<https://www.sigfox.es/faqs>>
- [38] Universitat Oberta de Catalunya. ¿Qué es NB-IoT. Consultado el 07 de abril de 2021, <<https://informatica.blogs.uoc.edu/que-es-nb-iot/>>
- [39] Mekki, K., Bajic, E., Chaxel, F., Meyer, F. (2019) A comparative study of LPWAN technologies for large-scale IoT deployment . The Korean Institute of Communication and Information Sciences
- [40] Semtech. LoRa Technology Is Connecting Our Smart Planet. Consultado el 09 de abril de 2021, <<https://www.semtech.com/lora/lora-applications>>
- [41] Heyuan Intelligence. What is LoRa Protocol? Consultado el 09 de abril de 2021, <<http://www.electricityiot.com/2019/03/18/what-is-lora-protocol/>>
- [42] Logitek. Caso de uso: LoRaWAN en industria. Consultado el 12 de abril de 2021, <<https://www.m2mlogitek.com/caso-de-uso-lorawan-en-industria/>>
- [43] Jedermann, R., Borysov, M., Hartgenbusch, N., Jaeger, S., Sellwig, M., & Lang, W. (2018) Testing Lora for food applications - Example application for airflow measurements inside cooled warehouses with apples. *Procedia Manufacturing*, 24, 284-289. [Consultado el 12 de abril de 2021. Disponible en: <<https://doi.org/10.1016/J.PROMFG.2018.06.026>>
- [44] Lee Y., Taberner T. (abril 2020). LoRa ofrece nuevas oportunidades en IoT Industrial - Una introducción a las tecnologías inalámbricas y sus aplicaciones en la automatización industrial. Advantech. Consultado el 13 de abril de 2021. Disponible en: <<moz-extension://e68403a1-6720-427f-8a1e-ab4b337ca29b/enhanced-reader.html?openApp&pdf=https%3A%2F%2Fwww.redeweb.com%2FAdvantech-Lora.pdf>>
- [45] Atim. Metering and Dry Contacts DINDxxx (2020). Consultado el 14 de abril de 2021. Disponible en: <http://www.atim.com/docs/ANGLAIS/ACW%20MODEMS/ACW-DINDxxx/ATIM_ACW-DINDxxx_UG_EN_V1.8.pdf>
- [46] Actility. Volvo Group implements an IoT tracking solution in its trucks plant, using Actility LoRaWAN GPS trackers and network management. Consultado el 15 de abril de 2021, <<https://www.actility.com/volvo-group-implements-in-its-trucks-plant-an-iot-tracking-solution-using-lorawan-gps-trackers-and-network-management-from-actility/>>

- [47] HiveMQ. Introducing the MQTT Protocol - MQTT Essentials: Part 1. Consultado el 15 de abril de 2021, <<https://www.hivemq.com/blog/mqtt-essentials-part-1-introducing-mqtt/>>
- [48] BehrTech. What is MQTT and Why You Need It in Your IoT Architecture. Consultado el 15 de abril de 2021, <<https://behrtech.com/blog/mqtt-in-the-iot-architecture/>>
- [49] Industrial Shields. Fundamentos de seguridad del MQTT en la automatización industrial. Consultado el 16 de abril de 2021, <https://www.industrialshields.com/es_ES/blog/blog-industrial-open-source-1/post/fundamentos-de-seguridad-del-mqtt-en-la-automatizacion-industrial-235>
- [50] OPC Foundation. Unified Architecture. Consultado el 16 de abril de 2021, <<https://opcfoundation.org/about/opc-technologies/opc-ua/>>
- [51] Revista Digital. Protocolo OPC UA. Características y aplicaciones en SCADA. Consultado el 16 de abril de 2021, <<https://revistadigital.inesem.es/gestion-integrada/protocolo-opc-ua-caracteristicas-y-aplicaciones/>>
- [52] Github. Client - Server Application. Consultado el 17 de abril de 2021, <https://madooei.github.io/cs421_sp20_homepage/client-server-app/>
- [53] Real Time Automation. An Introduction to OPC UA. Consultado el 17 de abril de 2021, <<https://www.rtautomation.com/technologies/opcu/>>
- [54] About SSL. The Ultimate Guide to SSL/TLS Client Authentication: Know How it Works. Consultado el 17 de abril de 2021, <<https://aboutssl.org/ssl-tls-client-authentication-how-does-it-works/>>
- [55] Atim. Indoor temperature and humidity ACW-TH-I. Consultado el 18 de abril de 2021, <https://www.atim.com/en/produits/catalogue/acw_en/sensor-temperature-humidity-indoor-acw-th-i/>
- [56] Atim. Flood detection ACW-WL. Consultado el 19 de abril de 2021, <https://www.atim.com/en/produits/catalogue/acw_en/flood-detection-sensor-acw-wl/>
- [57] Atim. Level measurement ACW-LVL. Consultado el 19 de abril de 2021, <https://www.atim.com/en/produits/catalogue/acw_en/level-measurement-acw-lvl/>
- [58] Browan (2020). Outdoor Micro Gateway Getting Started Guide with Basic Station & AWS IoT Core for LoRaWAN. Consultado el 20 de abril de 2021. Disponible en: <<https://www.browan.com/product/outdoor-micro-gateway/detail>>

- [59] Browan (2020). Browan ODU Installation Guide. Consultado el 21 de abril de 2021. Disponible en: <https://www.browan.com/product/outdoor-micro-gateway/detail>
- [60] Actility. ThingPark Enterprise Dashboard. Consultado el 22 de abril de 2021, <https://thingparkenterprise.eu.actility.com/tpe/#/dashboard>
- [61] Actility (28 de febrero de 2020). Actility ThingPark Enterprise On Customer Premises User Guide. Consultado el 22 de abril de 2021. Disponible en: <https://docs.thingpark.com/thingpark-enterprise/6.1/Content/SNMP-MIBs-TPE-documentation-library.htm>
- [62] Actility (9 de diciembre de 2020). Actility ThingPark Enterprise LRC-AS Tunnel Interface Developer Guide (LoRaWAN®). Consultado el 24 de abril de 2021. Disponible en: <https://docs.thingpark.com/thingpark-enterprise/6.1/Content/SNMP-MIBs-TPE-documentation-library.htm>
- [63] Node-RED. Editor Guide. Consultado el 25 de abril de 2021, <https://nodered.org/docs/user-guide/editor/>
- [64] Wonderware Iberia. ¿Qué es SCADA? Supervisory Control and Data Acquisition (Supervisión, Control y Adquisición de Datos). Consultado el 26 de abril de 2021, <https://www.wonderware.es/hmi-scada/que-es-scada/>
- [65] Actility. Actility ThingPark Gemtek ODU Macro V1.5 - 8 channels Software Installation and Operation Guide. Consultado el 28 de abril de 2021, Disponible en: https://repository.community.thingpark.com/documents?type=TPE_BS_DOC&vendor=gemtek&model=GMTK/US
- [66] Actility. Actility ThingPark X. Consultado el 28 de abril de 2021, https://docs.thingpark.com/thingpark-x/latest/Message/Downlink_Message/#original-message
- [67] Kepware. Kepware KEPServerEX. Consultado el 28 de abril de 2021. Disponible en: <https://www.kepware.com/getattachment/e1943820-ef3c-4932-b055-4ef2a80ab863/kepserverex-manual.pdf>
- [68] Semtech. The Future of 5G and LoRaWAN®: Friends or Foes?. Consultado el 20 de mayo de 2021. Disponible en: <https://blog.semtech.com/the-future-of-5g-and-lorawan-friends-or-foes/>

Apéndice A

Presupuesto

En este apéndice del proyecto se desglosa el coste necesario para la adquisición e instalación del hardware y software para el montaje de la red privada LoRaWAN desarrollada.

En el cuadro A.1 se muestra el coste de cada uno de los equipos utilizados. De igual manera, se indica una estimación de su coste de instalación.

En el cuadro A.2 se recoge el coste de todas las licencias necesarias para activar los programas utilizados en el proyecto.

Respecto al coste del personal requerido para desarrollar y desplegar esta red LoRaWAN, su aproximación queda recogida en el cuadro A.3.

Finalmente, en el cuadro resumen A.4 se indica la suma del coste del hardware y el software necesarios para implementar el prototipo de red LoRaWAN desarrollado en el presente proyecto.

Cuadro A.1: Coste del hardware y su instalación

	Precio material (€)	Cantidad	Coste instalación (€)	Total por dispositivo (€)
Nodo ACW/LW8-TH	145	1	20	165
Nodo ACW/LW8-LVL	349	1	20	369
Nodo ACW/LW8-WL	299	1	20	319
Gateway ODU Macro V1.5 - 8 channels	810	1	40	850
Total	1603	-	100	1703

Cuadro A.2: Coste del software

	Coste (€)
Licencia ThingPark Enterprise	1200
Licencia MQTT Client Driver (KEPServerEX)	409
Licencia IoT Gateway (KEPServerEX)	379
Licencia AVEVA Edge 2020	2740
Total Software	4728

Cuadro A.3: Coste del personal

	Tiempo (h)	Coste por tiempo (€/h)	Total por labor (€)
Desarrollo de la red	500	12	6000
Montaje y despliegue	100	12	1200
Total	600	-	7200

Cuadro A.4: Suma del coste del hardware, software y personal

	Coste (€)
Hardware	1703
Software	4728
Personal	7200
Total	13631