

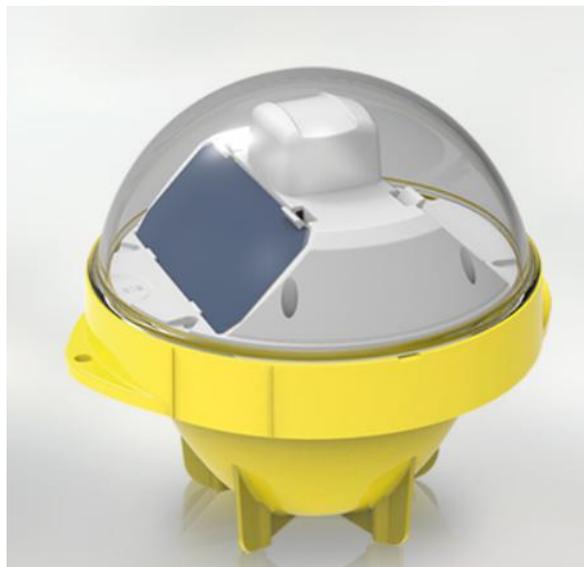
Tecnologías IIoT, hands-on y MVPs

Práctica: MVP para el caso transversal "depuradora"

Presentación

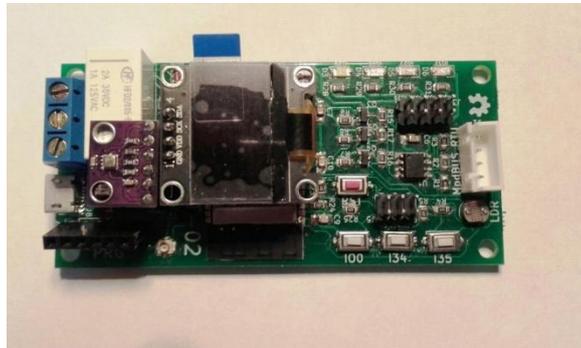
El objetivo principal de esta práctica es desarrollar un mínimo producto viable (MVP) de un sistema de IoT flotante, alimentado por batería de larga autonomía, que permita obtener lecturas del nivel del líquido en tanques, presión, temperatura, humedad relativa, nivel de luz, y en una evolución futura podría incorporar lecturas de sensores en contacto con el líquido como PH y otras informaciones provenientes del líquido de los tanques. El trabajo debe realizarse preferentemente en equipo, pudiendo también ser realizado de forma individual.

En el sector de la pesca hay disponibles diversos productos consistentes en boyas con conexión vía satélite para el seguimiento de bancos de peces y aplicaciones similares. Supondremos que la parte mecánica será resuelta por un equipo ajeno, así como un mecanismo de igualamiento de la presurización del interior con el exterior mediante una válvula de apertura puntual antes de realizar las lecturas de presión, temperatura u otras.



En la figura puede observarse una boya del fabricante Satlink
(<https://industriaspesqueras.com/noticia-66393-sec-Ferias%20y%20Exposiciones>).

En nuestro caso la tecnología de comunicaciones inalámbricas a utilizar será a utilizar será Wi-Fi estándar, y utilizaremos la placa de prototipaje IoT-02, de hardware abierto, basada en el procesador ESP32:



Objetivos y competencias

Los **objetivos** de esta actividad son los siguientes:

- 1) Conocer la conexión de sensores a microcontroladores de escala mediana con capacidades de conexión inalámbrica, lo que incluye la programación del microcontrolador y la comunicación con el mismo vía MQTT, tanto con gemelo digital como con sistema físico.
- 2) Conectar sistemas embebidos a PCs industriales o al *cloud*, almacenando datos de las lecturas en base de datos como series temporales, lo que incluye bases de datos SQL y hoja de cálculo en la nube.
- 3) Utilizar sistemas de software para IoT como Node-RED y Grafana para monitorizar y visualizar los datos de las lecturas mediante dispositivos con navegador.
- 4) Combinar sistemas de IoT con redes sociales como Telegram o otras para facilitar la interacción con las personas.
- 5) Conocer y experimentar con un reto ligado a un caso transversal, en este caso el de la depuradora.

Requisitos previos: Tener conocimientos básicos de programación de ESP32 en entorno Arduino, de comunicación con MQTT, de almacenamiento de datos y de representación de datos (*dashboards*).

Realizad una memoria sintética en PDF adjuntando capturas de pantalla y acompañadla de un fichero zip con los programas (exportaciones de Snap! y flujos de Node-RED).

Ejercicio 1 (Obtención de datos)

Se dispone de un firmware para la placa IoT-02 (https://github.com/jordibinefa/loT-02/tree/master/codes/loT-02_mqtt_sync_upcSchool_broker1). El archivo de cabecera `IoT-02_mqttTopics.h` define los temas (topics) MQTT disponibles en la placa. Los que contienen en su alias (define) REQUEST o REQ son para publicar en ellos y el resto para subscribirse a ellos. Los que contienen SYNC son de tipo síncrono. En particular, publicando una petición síncrona (request) en `/<MAC>/sync/jsonIn` se obtiene una respuesta como:

```
{ "Tx100": 2646, "RHx100": 3627, "Px100": 100503, "Ax100": 6858, "bI00": "0", "bI34": "0", "bI35": "0", "LDR": 2001 }
```

También se puede obtener el JSON anterior de forma asíncrona subscribiendo Snap! o Node-RED a `/<MAC>/jsonIn` y publicando en `/<MAC>/jsonInReq`.

Se pide que realicéis cuatro programas:

1. Obtener los datos con Snap! de forma asíncrona (bloques publish y subscribe)
2. Obtener los datos con Snap! de forma síncrona (bloque request-response)
3. Obtener los datos con Node-RED de forma asíncrona (nodos publish y subscribe)
4. Obtener los datos con Node-RED de forma síncrona (nodo subflow MQTT sync)

Programad si es necesario la placa de electrónica embebida IoT-02, indicando en caso los datos de conexión de vuestro punto de acceso Wi-Fi, y verificad los datos relativos al broker MQTT que vais a utilizar (podéis optar entre broker seguro o broker no seguro):

Broker no seguro
Datos del <i>broker</i> :
Servidor: vps656540.ovh.net
Puerto MQTT: 1883
Puerto Websockets (no seguro): 8081
Puerto Websockets (seguro): 8082
Usuario: user
Contraseña: pass

Ejercicio 2 (Almacenamiento de datos)

Ampliad los programas en Node-RED del ejercicio 1 para almacenar datos, concretamente las lecturas de presión, temperatura y nivel de luz.

Se pide que realicéis cuatro programas:

1. Obtener los datos con Node-RED de forma asíncrona y almacenar los datos en una base de datos MySQL¹
2. Obtener los datos con Node-RED de forma asíncrona y almacenar los datos en una hoja de cálculo en la nube (G Sheets)
3. Obtener los datos con Node-RED de forma síncrona y almacenar los datos en una base de datos MySQL
4. Obtener los datos con Node-RED de forma síncrona y almacenar los datos en una hoja de cálculo en la nube (G Sheets)

Se propone que antepongáis el prefijo `"m" + puerto + "_"` al nombre de las tablas de la base de datos y al de las pestañas de la hoja de cálculo en el caso de utilizar la suministrada como ejemplo.

En cuanto a G Sheets se pide que implementéis vuestra propia hoja de cálculo. Se puede también utilizar una pestaña de la hoja de cálculo suministrada como ejemplo, aunque esta opción recibirá menor puntuación.

Ejercicio 3 (Representación de datos)

Representar gráficamente las lecturas de presión, temperatura, humedad relativa y nivel de luz almacenadas en la base de datos MySQL.

Se pide que implementéis tres representaciones:

1. Con un gráfico en la hoja de cálculo.
2. En un panel de control (*dashboard*) de Node-RED.
3. En un panel de control (*dashboard*) de Grafana.

Utilizad los usuarios de Grafana disponibles en el VPS del máster, y nombrar los paneles de control (*dashboards*) según la regla del prefijo `"m" + puerto + "_"`.

¹ En nuestro caso utilizaremos MariaDB, que es un fork open source del proyecto original de MySQL (MariaDB está impulsado por Google y MySQL está en la órbita de Oracle)

Ejercicio 4 (Conexión con redes sociales)

Realizad un bot de Telegram que combinado con Node-RED permita la consulta en tiempo real de la presión, temperatura, humedad relativa y nivel de luz.

El acceso concurrente al microcontrolador de varias peticiones puede colapsarlo. Implementad una solución que permita la consulta concurrente de múltiples usuarios del valor del último minuto de la presión, temperatura, humedad relativa y nivel de luz.

Elaborad un enlace (link) que permita por HTTPS consulta de la presión, temperatura, humedad relativa y nivel de luz del último minuto. Este enlace es susceptible de ser pasado y ejecutado por otras redes sociales como WhatsApp, Instagram o X, o bien mediante un enlace en un correo electrónico o en una página web.

Utilizad Node-RED como pasarela (gateway) universal de protocolos IoT.

Ejercicio 5 (Caso transversal - Depuradora)

El sensor de presión se puede utilizar para medir la altura en la que está la placa IoT-02, la cual se puede medir calculando la diferencia entre las lecturas entre dos puntos. En un escenario ideal, tendríamos de dos placas IoT-02, una colocada a la altura del suelo del depósito y la otra colocada en la boya flotante, de manera que la altura se calcularía por la diferencia de las dos lecturas.

Como sólo disponemos de una placa, para calcular la altura, deberemos realizar una primera lectura con la placa en el suelo. Para ello pulsad el botón I34 de la placa IoT-02 para pedir al Node-RED que lea la presión a nivel del suelo. Salvad este dato como variable de flujo o global de Node-RED, y utilizadla para el cálculo de la diferencia con la IoT-02 y ver cómo va evolucionando en tiempo real. La presión de la placa a la altura del suelo va variando con el tiempo, pero supondremos que la variación de ésta mientras dure el experimento, no es significativa.

Alimentar la placa con un cable lo suficientemente largo o con un *powerbank* de teléfono móvil para realizar el experimento.

Se pide que incorporéis en el *dashboard* del ejercicio 3 la visualización de la altura de la placa IoT-02 respecto al suelo en tiempo real. Valorad la precisión del sistema y proponed un sistema de redondeo para aplicar en embalses o depósitos de depuradoras.