



### ALERTAS MÉDICAS — Descripción del escenario

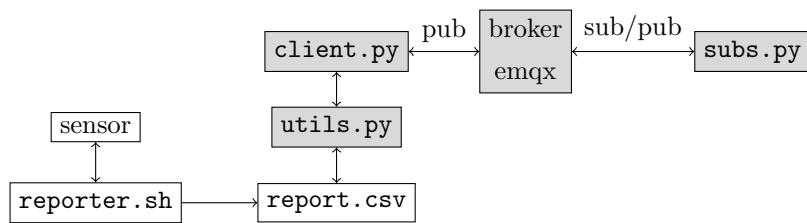


Figura 1: escenario de la práctica.

El escenario para el mini-reto de generación de alertas médicas se basa en un sistema que simula la monitorización de constantes vitales de un paciente. En este entorno, se emplean varios componentes que interactúan entre sí utilizando el protocolo MQTT para la transmisión de datos. La arquitectura del sistema se representa en el esquema proporcionado.

- **Sensor:** Representa el dispositivo de monitorización biomédica, encargado de medir las constantes vitales del paciente (por ejemplo, ritmo cardíaco, frecuencia respiratoria, SpO<sub>2</sub>, temperatura).
- **reporter.sh:** Es un script que simula el comportamiento del sensor. Recoge los datos del paciente y los escribe en un archivo CSV, `report.csv`, que actúa como la base de datos simulada del paciente.
- **client.py:** Este script en Python actúa como publicador de datos. Lee las constantes vitales desde el archivo `report.csv` y las publica a través del protocolo MQTT al broker EMQX. Publica los datos en el topic correspondiente, como `vitals/hr`, `vitals/spo2`, etc.
- **utils.py:** Contiene funciones auxiliares que permiten la manipulación y conversión de los datos de `report.csv` en formato adecuado para la publicación MQTT. Una de estas funciones es `line_to_dict()`, que convierte cada línea del archivo CSV en un diccionario Python.

- **broker EMQX:** Actúa como el servidor MQTT, recibiendo y distribuyendo los mensajes publicados por el cliente y reenviándolos a los nodos suscriptores. El broker se encarga de gestionar las conexiones y la entrega de los mensajes a los suscriptores.
- **subs.py:** Este script se utiliza para la suscripción a los topics de MQTT. El suscriptor se conecta al broker EMQX, recibe los mensajes publicados por el cliente y puede procesar los datos recibidos, generando alertas si se detectan anomalías.