

Desarrollo de Sistemas IoT

Fundamentos y Arquitectura de Red

Edward Goyeneche y Gustavo Osorio

Departamento de Ingeniería Eléctrica, Electrónica y Computación
Universidad Nacional de Colombia - Sede Manizales

2025-08-29



Objetivos de la sesión

- Comprender el concepto básico de un sistema IoT.
- Ejecutar un ejemplo práctico con MQTT usando el repositorio `Demo_IoT_MQTT_SERVER`.
- Relacionar el ejemplo con la arquitectura de referencia IoT (ISO/IEC 30141).
- Identificar los componentes clave: nodo, gateway y servidor.

Motivación

¿Por qué aprender IoT?

- El Internet de las Cosas conecta el mundo físico con el digital.
- Está en el corazón de las transformaciones actuales:
 - Ciudades inteligentes
 - Industria 4.0
 - Energía y redes inteligentes
 - Salud digital y telemedicina
- Entender IoT = entender cómo se construyen sistemas que impactan la vida cotidiana.

Motivación

Impacto social y económico

- IoT es uno de los sectores más dinámicos de las TIC.
- Permite nuevos modelos de negocio: *“smart products” y servicios basados en datos.*
- Aporta a la eficiencia energética, sostenibilidad y seguridad.
- Genera oportunidades de emprendimiento tecnológico en regiones como el Eje Cafetero.

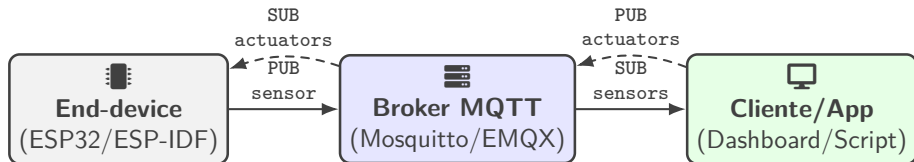
Motivación

Conexión con estándares

Según ISO/IEC 30141:2024, un sistema IoT tiene dos características esenciales:

- 1 Componentes interconectados en red digital (muchos-a-muchos).
- 2 Al menos un componente que interactúa con el mundo físico (sensor o actuador).

Arquitectura mínima IoT



Escenario: nodo publica mediciones; la aplicación se suscribe; el broker enruta mensajes. (Patrón *publish/subscribe*).

Componentes y responsabilidades

End-device

- Sensado/actuación.
- MQTT *client*.
- QoS 0/1 según criticidad.
- Reconexión y retención de estado.

Broker MQTT

- Enrutamiento por tópico.
- Autenticación/ACL.
- Retained messages / LWT.
- Escalabilidad (n clientes).

Cliente/App

- Visualizar y/o decidir.
- Suscripción a tópicos.
- Persistencia/alertas.
- Envío de comandos.

Flujo de datos MQTT (mínimo viable)

- 1 **Conexión:** cada cliente (nodo o app) se conecta al broker (CONNECT).
- 2 **Autenticación:** usuario/clave o certificado (TLS).
- 3 **Suscripción:** la app envía SUBSCRIBE a sensors/#/temp.
- 4 **Publicación:** el nodo envía PUBLISH en sensors/room1/temp con QoS=1.
- 5 **Entrega:** el broker reenvía a todos los clientes suscritos (filtro por tópicos).
- 6 **Opcional:** comandos a actuadores (actuators/room1/fan).

Buenas prácticas: nombres de tópico jerárquicos, payload JSON/CBOR, timestamps, IDs únicos.

Mapecto a la referencia ISO (arquitectura mínima)

- **IoT component:**

- End-device (ESP32) con interfaz de red y capacidad de sensado/actuación.
- Broker MQTT y Cliente/App como componentes con interfaces de red.

- **IoT system:** composición de componentes conectados por red digital.

- **Interacción con el mundo físico:** sensor/actuador en el end-device.

- **IoT environment:** la red y servicios de soporte (DNS, NTP, PKI, almacenamiento).

Con tres piezas (nodo, broker, app) ya podemos ejemplificar un *sistema IoT* completo.

Seguridad mínima en la arquitectura

Controles esenciales

- **Cifrado en tránsito:** MQTT sobre TLS (mqtts).
- **Autenticación:** usuario/clave o certificados cliente.
- **Autorización:** ACL por patrón de tópico.
- **LWT:** Detectar desconexión inesperada de nodos.

Convenciones de tópicos

- `sensors/{site}/{dev}/{metric}`
- `actuators/{site}/{dev}/{cmd}`
- `sys/{dev}/status` (LWT)

Extensiones típicas (de mínimo viable a PoC)

- **Persistencia:** base de datos temporal/time-series (InfluxDB/Timescale).
- **Gestión:** registro de dispositivos, provisión de credenciales, OTA.
- **Observabilidad:** logs, métricas, trazas de red (broker + nodos).
- **Escalado:** balanceo, clúster de broker, particionado por tópicos.
- **Integración:** puente a HTTP/REST, gRPC, o plataforma (ThingsBoard, etc.).

Ejemplo: riego automático con MQTT (pub/sub)

Nodo IoT (ESP32 + sensor)

Publica cada minuto la humedad del suelo.

Dashboard/App

Se suscribe a la humedad, grafica y decide riego.

Actuador (válvula)

Se suscribe a un tópico de control para abrir/cerrar.

Flujo

- 1 ESP32 **publish** →
/finca/suelo/humedad
- 2 App **subscribe** ←
/finca/suelo/humedad
- 3 App decide y **publish** →
/finca/riego/cmd
- 4 Válvula **subscribe** ←
/finca/riego/cmd

Tópicos y payloads (ejemplos)

Lectura de sensor

```
Topic: /finca/suelo/humedad
Payload: { "valor": 40.2,
"unidad": "%RH", "ts":
1712345678 }
```

Comando a actuador

```
Topic: /finca/riego/cmd
Payload: { "accion": "ON",
"duracion_s": 120 }
```

Opcional (estado)

```
Topic: /finca/riego/estado
Payload: { "estado": "ABIERTO" }
```

Relación con IoT RA (ISO/IEC 30141)




Definición esencial

- Un **sistema IoT** combina:
 - **Red digital muchos-a-muchos** entre componentes, y
 - **Interacción con el mundo físico** (sensado/actuación).

Mapeo del demo MQTT

- **IoT component**: ESP32 (sensor/actuador) con interfaz de red.
- **IoT system**: nodo (ESP32) + broker MQTT + cliente(s).
- **IoT environment**: la red (Wi-Fi/LAN) y servicios de soporte (DNS, NTP, PKI, almacenamiento).

Fundamentos

-  **Composabilidad:** módulos que se combinan (dispositivo, broker, app, almacenamiento).
-  **Heterogeneidad:** variedad de HW/SW y protocolos (MQTT, CoAP HTTP; Wi-Fi, Ethernet, LTE).
-  **Distribución:** nodos y gateways en campo, conectados por redes diversas

IoT \neq IT “pura”

-  **Sensado/actuación:** el sistema percibe y modifica el entorno.
- **Ciclo físico–digital:** telemetría \rightarrow decisión \rightarrow comando.
- **Ejemplo demo:** sensor (ESP32) \rightarrow MQTT \rightarrow cliente \rightarrow (opcional) actuador.

Viewpoints de arquitectura

- **Fundacional:** ¿qué es IoT y qué lo diferencia? Conceptos, límites del sistema y relaciones básicas.

Discusión

- ¿Qué preocupaciones tiene un **ingeniero** del sistema?
- ¿Y un **administrador de red**?
- ¿Y un **usuario final**?
- ¿Y un **diseñador de la arquitectura**?

Viewpoints de arquitectura




- **Fundacional:** ¿qué es IoT y qué lo diferencia? Conceptos, límites del sistema y relaciones básicas.
- **Uso:** ¿cómo interactúan usuarios y sistemas? Casos de uso, roles, escenarios de operación.

Discusión

- ¿Qué preocupaciones tiene un **ingeniero** del sistema?
- ¿Y un **administrador de red**?
- ¿Y un **usuario final**?
- ¿Y un **diseñador de la arquitectura**?

Cierre y reflexión

Lo aprendido hoy

-  **MQTT**: patrón *pub/sub*, tópicos, QoS básico.
-  **Arquitectura mínima**: nodo (ESP32) → broker → cliente.
-  **IoT RA (ISO/IEC 30141)**: *component/system/environment*.

Exit ticket (~1 min)

- 1 Una frase: *¿qué idea te llevas de hoy?*
- 2 Un riesgo técnico que ves en el *demo*.
- 3 Un cambio que harías para la próxima sesión.

Pregunta final

¿Qué añadirías (seguridad, persistencia, gestión, escalado, ...) para acercarte a producción?

Tarea

Herrero, cap. 1: anota 3 conexiones con el *demo* y 1 pregunta para discusión.