

## Diagramas de estados para la conversión del protocolo MQTT-SN a MQTT utilizando UML

### State diagrams for the conversion of the MQTT-SN protocol to MQTT using UML

Carlos Panchi<sup>1</sup> <https://orcid.org/0000-0001-9624-6379>,  
Carlos Egas<sup>1</sup> <https://orcid.org/0000-0002-3540-9768>

<sup>1</sup>Escuela Politécnica Nacional, Quito, Ecuador

[carlos.panchi@epn.edu.ec](mailto:carlos.panchi@epn.edu.ec), [carlos.egas@epn.edu.ec](mailto:carlos.egas@epn.edu.ec)



Esta obra está bajo una licencia internacional  
Creative Commons Atribución-NoComercial 4.0.

Enviado: 2022/07/02

Aceptado: 2022/09/22

Publicado: 2022/11/30

#### Resumen

Las implementaciones del convertidor de protocolos MQTT-SN a MQTT han considerado que MQTT-SN opera sobre capa de red. En redes inalámbricas de sensores con topología lineal, existe una sola ruta y los nodos inalámbricos tienen una única interface, por lo tanto, no serían necesarios protocolos de enrutamiento. En este artículo se presenta el diagrama de estados del convertidor de protocolos para ser utilizado en una red con MQTT-SN encapsulado en capa de enlace como un insumo para su posterior implementación. Para desarrollar el diagrama de estados se utilizó la metodología del lenguaje de modelado unificado, los diagramas de estados contienen procesos que permiten la codificación del convertidor de protocolos. Se realizó una revisión teórica y funcional de los protocolos MQTT y MQTT-SN para conocer los mensajes incluidos en cada uno, así como de los campos de mensaje que interactúan en el proceso de conversión. El evento asociado con la llegada de mensajes al convertidor, activan los cambios de estados y desencadena una serie de procesos que concluyen en la generación de un mensaje convertido. Para encontrar inconsistencias o solucionar problemas de lógica en los estados y procesos obtenidos en el diagrama de estados, se utilizó la herramienta UPPAAL.

**Palabras clave:** convertidor, estados, UML, diagrama, mensajes.

#### Abstract

Implementations of the MQTT-SN to MQTT protocol converter have considered that MQTT-SN operates over the network layer. In wireless sensor networks with linear topology, there is

**Sumario:** Introducción, Trabajos relacionados, Metodología, Resultados y Discusión y Conclusiones.

**Como citar:** Panchi, C. & Egas, C. (2022). Diagramas de estados para la conversión del protocolo MQTT-SN a MQTT utilizando UML. *Revista Tecnológica - Espol*, 34(3), 84-99.  
<http://www.rte.espol.edu.ec/index.php/tecnologica/article/view/953>

a single path and wireless nodes have a single interface, therefore, no routing protocols would be necessary. This paper presents the state diagram of the protocol converter to be used in a network with MQTT-SN encapsulated in the link layer as an input for further implementation. To develop the state diagram, the methodology of the unified modeling language was used; the state diagrams contain processes that allow the codification of the protocol converter. A theoretical and functional review of the MQTT and MQTT-SN protocols was carried out to learn about the messages included in each one, and the message fields that interact in the conversion process. The event associated with the arrival of messages at the converter activates state changes and triggers a series of processes that conclude in the generation of a converted message. To find inconsistencies or solve logic problems in the states and processes obtained in the state diagram, the UPPAAL tool was used.

**Keywords:** converter, states, UML, diagram, messages.

### Introducción

El constante desarrollo tecnológico en las redes y telecomunicaciones, está relacionado con las comunicaciones inalámbricas y el internet de las cosas. Las comunicaciones inalámbricas han propiciado el desarrollo de las redes inalámbricas de sensores (WSN) (Eghonghon Ukhurebor et al., 2021), aplicadas a las redes industriales (Queiroz et al., 2017) y redes sensoras actuadoras (Kumar S. et al., 2014) lo cual genera un crecimiento exponencial de su utilización, gracias a la gran variedad de aplicaciones en las cuales esta tecnología puede ser utilizada. Gran parte de las aplicaciones de las redes inalámbricas de sensores, están en el área del internet de las cosas (IoT) (Adepoju, 2022) debido a que facilita la automatización y monitoreo con dispositivos que tienen recursos limitados en su capacidad de cómputo y de energía.

Las redes utilizadas en IoT, permiten la transportar los datos entre el nodo sensor y el colector de datos. En algunas aplicaciones, la información que circula en la red se entrega a los colectores de datos según su importancia. Es decir, se tiene una comunicación centrada en los datos que se adapta a redes con nodos que tienen limitaciones de procesamiento.

En la actualidad se pueden listar varios protocolos usados por las aplicaciones de IoT, tales como: el protocolo de aplicación restringida (CoAP) (Serianni & De Rango, 2022), el protocolo de mensajería basado en suscripciones y publicaciones (MQTT) y el protocolo MQTT para redes sensores (MQTT-SN) (Al Rasyid et al., 2019). Cada uno de ellos tiene características propias y una arquitectura de mensajería para diferentes tipos de aplicaciones de IoT, que tienen como premisa general, utilizar efectivamente los recursos que usualmente son limitados en los nodos de red. Una comunicación centrada en los datos es el modelo de mensajería Publicación/Suscripción (Pub/Sub), ampliamente usada y escalable.

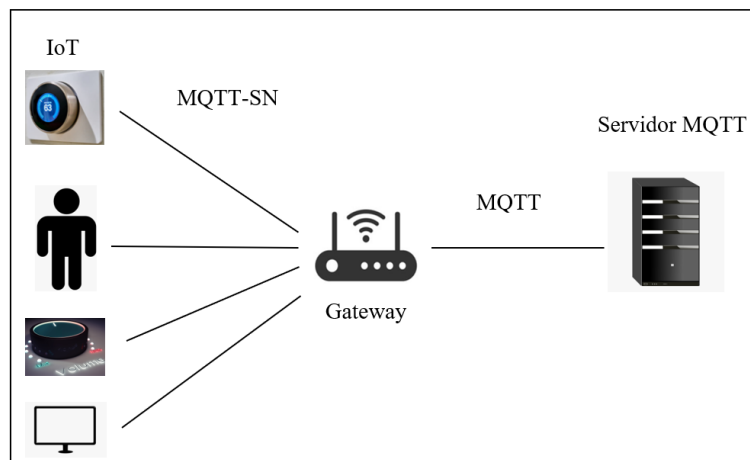
En redes IoT Figura 1, se utiliza el protocolo MQTT conjuntamente con la extensión creada para redes inalámbricas de sensores denominada MQTT-SN creado para ser ejecutado en nodos con limitaciones de cálculo y de energía, y que tiene garantizado su operabilidad con MQTT. Para que coexistan estos dos protocolos se requiere un proceso de conversión de protocolos que se debe realizar en un nodo interno de la red para garantizar su interoperabilidad. El protocolo MQTT-SN es usado por los nodos inalámbricos de sensores desplegados en la red y MQTT es utilizado por el servidor (Broker) que acepta mensajes publicados por clientes y los difunde entre los clientes suscritos.

Las redes inalámbricas de sensores son implementadas con varias arquitecturas de red, asociadas con los siguientes protocolos: 6LowPAN (Chen et al., 2011), WirellesHart (Luo et

al., 2021), ISA100 (Raptis et al., 2020), ZigBee (Baquer et al., 2018), los cuales utilizan el protocolo IEEE 802.15.4 como protocolo de enlace.

**Figura 1**

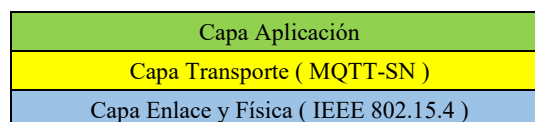
*Red MQTT-SN y MQTT*



El Grupo de Investigación de Internet de Todas las Cosas de la Escuela Politécnica Nacional, trabaja en la propuesta de una nueva arquitectura de red, sin la capa de red, para ser utilizada en redes con topología lineal multisalto (Egas & Gil-Castiñeira, 2020), considerando que en topologías lineales no existe enrutamiento al existir un solo camino entre fuente y destino (Acosta et al., 2021). De igual manera, no existe conmutación en el nodo sensor inalámbrico debido a que tiene únicamente una interface por la cual recibe y transmite tramas. Por lo tanto, en la arquitectura de red propuesta Figura 2 se requiere que los mensajes del protocolo MQTT-SN se encapsulen directamente en las tramas del protocolo IEEE 802.15.4.

**Figura 2**

*Arquitectura para WSN con topología lineal*



Con este antecedente, es necesario disponer de un Gateway para permitir la convertibilidad entre MQTT-SN y MQTT, en redes donde los mensajes MQTT-SN son encapsulados en el protocolo IEEE 802.15.4.

Por lo tanto, el objetivo de este artículo es obtener el diagrama de estados para especificar la conversión del protocolo MQTT-SN a MQTT mediante el lenguaje unificado de modelado (UML) (Mura & Sami, 2008). Los diagramas obtenidos son el insumo principal para la codificación e implementación de los procesos de conversión del protocolo MQTT a MQTT-SN que opera sobre IEEE 802.15.4.

El presente trabajo, sigue la siguiente estructura: en la sección 2 se presenta una breve revisión de investigaciones relacionadas con el problema, en la sección 3 se menciona la metodología utilizada para la obtención del diagrama de estados, en la sección 4 se presentan los resultados y discusión del diagrama de estados propuesto. Finalmente, en la última sección, se plantean las conclusiones, así como futuros trabajos a realizar.

### Trabajos relacionados

Nuevas propuestas se han presentado, para mejorar el desempeño de MQTT-SN y por lo tanto facilitar la creación de nuevas aplicaciones. En (Fontes et al., 2020) se proporciona un mecanismo de calidad de servicio utilizando MQTT-SN para mejorar el tiempo de llegada de los mensajes en la red. Una nueva arquitectura de seguridad se propone en (Park & Nam, 2020) para mejorar la seguridad de redes inalámbricas de sensores con MQTT-SN, con resultados de nuevas políticas de seguridad para los suscriptores. En (Al Rasyid et al., 2019) los autores analizan el comportamiento de MQTT-SN en aplicaciones de ciudades inteligentes. En todos los casos, el protocolo MQTT-SN opera sobre nivel de red y los Gateway utilizados o implementados operan con esta característica.

El desarrollo de aplicaciones que utilizan redes inalámbricas de sensores ha incentivado a diferentes empresas para desarrollar aplicaciones para realizar la conversión del protocolo MQTT-SN a MQTT, por ejemplo el desarrollado por Paho (Paho, 2021) y EMQX (EMQX, 2020) entre otros. Sin embargo, estos convertidores de protocolos son implementados sobre el nivel de red.

### Metodología

El Lenguaje Unificado de Modelado (UML) (Koc et al., 2021), es el lenguaje y la metodología utilizada en este trabajo para modelar el funcionamiento del convertidor de protocolos de MQTT-SN a MQTT. Como resultado del modelamiento se obtienen los diagramas de estados que permitan posteriormente realizar la codificación e implementación del convertidor.

### Protocolo MQTT

El protocolo MQTT (MQTT, 2022) es el protocolo de transmisión de mensajes basado en el modelo publicación/suscripción, el cual está diseñado para operar en dispositivos provistos de limitadas velocidades de transmisión. El protocolo trabaja sobre TCP/IP u otros protocolos que permitan conexiones bidireccionales, ordenadas y sin pérdidas. El principal objetivo de MQTT es minimizar la utilización de energía en los nodos y el ancho de banda requerido por los dispositivos, con cierta fiabilidad en la red lo cual permite su uso en IoT o en conexiones máquina a máquina.

Los componentes requeridos por el protocolo MQTT para su operación son los siguientes:

- **Broker:** También denominado servidor, es el componente central que actúa como intermediario para recopilar los datos que transmiten los nodos publicadores y luego direccionar los mensajes a los nodos suscriptores que solicitan la información según el tópico del contenido.
- **Cliente:** Nodo que forma parte de la red y puede cumplir las funciones de publicador o suscriptor.
- **Publicador:** Nodo cliente encargado de la transmisión de información hacia el Broker sobre un tópico de interés determinado.
- **Suscriptor:** Nodo cliente que recibe la información del Broker sobre un tópico de interés determinado.
- **Tópico:** Es el tema al que los nodos clientes pueden suscribirse para recibir información.

Para conseguir la comunicación entre los componentes de red, el protocolo MQTT hace uso de varios mensajes de control para facilitar el intercambio de información.

En la Figura 3 se presenta la estructura del mensaje MQTT que contiene información que debe ser encapsulada en MQTT-SN.

**Figura 3**

*Estructura del mensaje MQTT (MQTT, 2022)*

Long. Campo (bit)	0	1	2	3	4	5	6	7
Byte 1	Tipo de Mensaje			DUP		QoS		Retención
Byte 2	Longitud Restante (1 - 4 Bytes)							
Byte 3	Opcional: Cabecera de Longitud Variable							
.								
Byte n								
Byte n+1	Opcional: Carga útil de Mensaje de Longitud Variable							
.								
Byte m								

### Protocolo MQTT-SN

El protocolo MQTT-SN (Stanford-Clark & Truong, 2013) es una extensión del protocolo MQTT, basado en el modelo de publicación/suscripción, para ser utilizado específicamente en redes inalámbricas de sensores (WSN). El objetivo de MQTT-SN es extender el alcance de MQTT en redes que no operen con la arquitectura TCP/IP, ya que esta arquitectura de red utiliza UDP en los dispositivos de red que no necesitan una conexión permanente. MQTT-SN es considerado como un protocolo ligero, adaptado para ser ejecutado en redes con nodos sensores que operan con un bajo ancho de banda, longitud de mensajes cortos, bajo consumo de batería y con limitada capacidad de procesamiento y almacenamiento.

En el protocolo MQTT-SN para su funcionamiento, requiere de los siguientes elementos:

- Cliente: Son los nodos que se encuentran en la red inalámbrica de sensores y al igual que en MQTT pueden operar como publicador o suscriptor. Para poder comunicarse con el Bróker MQTT requieren un Gateway.
- Gateway: Sirve de enlace entre los nodos cliente que utilizan el protocolo MQTT-SN y el Broker que usa el protocolo MQTT, por lo que este nodo es responsable de la conversión de protocolos.
- Forwarder: Es un nodo encargado de reencapsular las tramas MQTT-SN en otras tramas iguales, no realizan ningún cambio y las envían directamente al Gateway MQTT-SN para que se realice la conversión de protocolos que posteriormente se envían al Broker.

El nodo Gateway MQTT-SN puede operar de dos maneras en la red:

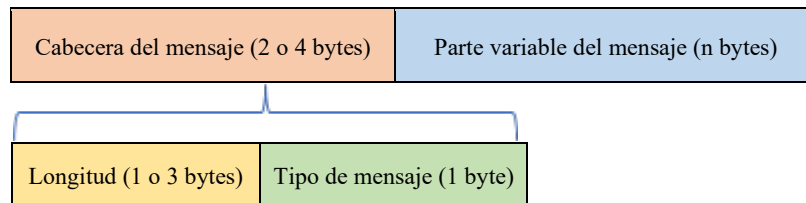
- Gateway transparente: El nodo administra una puerta de enlace para cada cliente por lo que la conversión de protocolos se realiza individualmente para cada flujo MQTT-SN generado en la red inalámbrica de sensores.
- Gateway agregado: El nodo posee solo una conexión MQTT con el Broker y administra la misma para dar servicio a un grupo de nodos cliente. La operación de

una arquitectura con Gateway agregado es complejo debido a que se debe mantener conexiones simultáneas con el cliente.

Al igual que en el protocolo MQTT, MQTT-SN tiene mensajes de control que son utilizados por los nodos clientes para establecer una comunicación con el Gateway. La estructura del mensaje MQTT-SN es diferente con relación al protocolo MQTT, tal como se muestra en la Figura 4.

**Figura 4**

*Estructura del mensaje MQTT – SN*



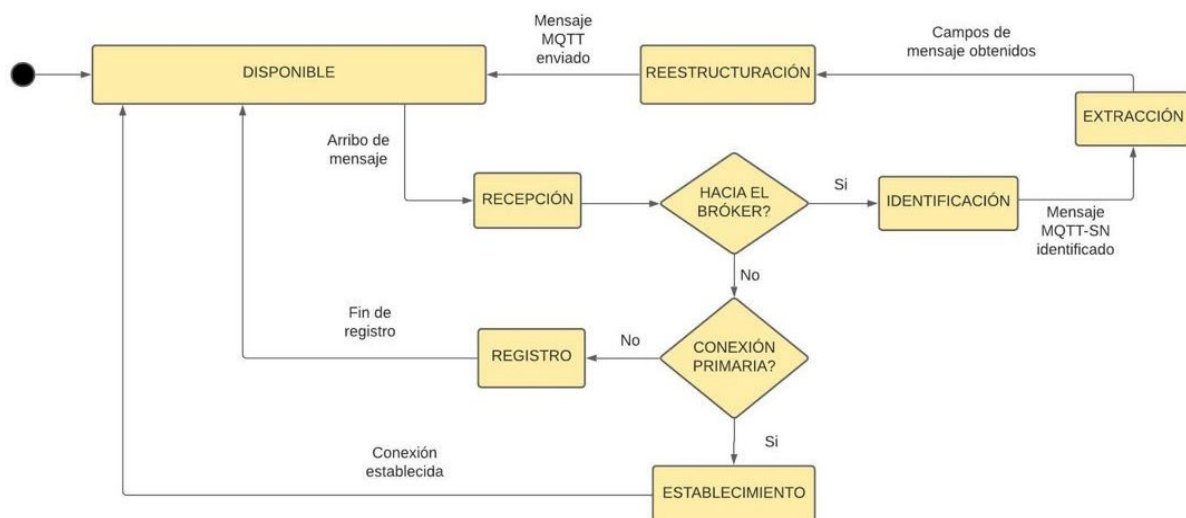
Los nodos de red requieren establecer una comunicación con el Bróker ya que es el encargado de recibir, guardar y reproducir la información transmitida por los nodos sensores. Debido a que el Bróker utiliza el protocolo MQTT, el nodo Gateway es el componente que funciona como interfaz entre ambos protocolos y realiza la conversión de MQTT-SN a MQTT. Debido a que MQTT-SN fue desarrollado a partir del protocolo MQTT para ser utilizado en redes inalámbricas de sensores, ambos protocolos guardan una similitud en sus mensajes a pesar de que tienen formatos de mensaje diferentes.

## Resultados y Discusión

Para que el Bróker pueda recibir los datos del nodo sensor encapsulados en MQTT-SN, el nodo sensor debe establecer en primer lugar una comunicación con el Gateway para que este convierta los mensajes de control al protocolo MQTT y puedan ser procesados por el Bróker. En la Figura 5 se observa el diagrama de estados diseñado en este trabajo, el cual representa el funcionamiento del convertidor de protocolos.

**Figura 5**

*Diagrama de estados del convertidor de protocolos MQTT-SN a MQTT*



El diagrama se conforma de diferentes estados, cada uno implica uno o más procesos que contribuyen a la funcionalidad del convertidor de protocolos. A continuación se procede a explicar cada uno.

### Estado DISPONIBLE

Es el estado principal del diagrama, en el cual el nodo Gateway acepta mensajes provenientes de la red y está apto para establecer una comunicación. Si se presenta un evento de arribo de mensaje se procede a realizar una transición al estado de RECEPCION. Se considera al estado DISPONIBLE, como base del desarrollo del diagrama del convertidor de protocolos ya que, al finalizar los procesos en los estados de REGISTRO, ESTABLECIMIENTO y REESTRUCTURACION se realiza una transición hacia el estado DISPONIBLE.

### Estado RECEPCION

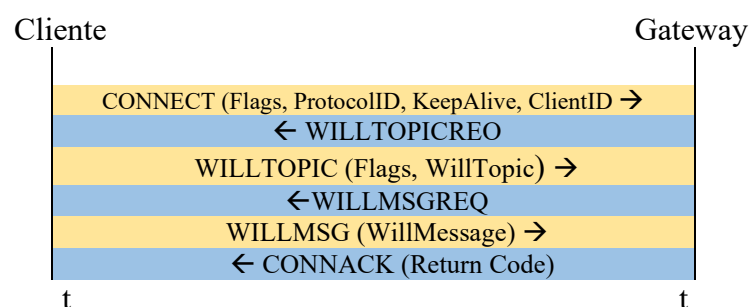
Si el Gateway se encuentra en estado DISPONIBLE y se presenta el evento de arribo de un mensaje, se realiza la transición al estado RECEPCION. En el presente estado el Gateway recibe el mensaje y decide si es un mensaje dirigido al Bróker MQTT o si es dirigido a sí mismo para establecer una conexión con un nodo sensor. Si el mensaje está dirigido hacia el Bróker se realiza una transición hacia el estado IDENTIFICACIÓN, caso contrario se cambia al siguiente pseudoestado de opción que divide las conexiones nodo sensor al Gateway en los estados de ESTABLECIMIENTO y REGISTRO los cuales realizan procesos de intercambio de mensajes.

### Estado ESTABLECIMIENTO

En este estado se realiza la conexión primaria con el nodo sensor, ya que previo a cualquier transmisión hacia el Bróker, se debe establecer una conexión con el Gateway con los mensajes que se muestran en la Figura 6.

**Figura 6**

*Conexión con el Bróker*



### Proceso de convertibilidad

El proceso de convertibilidad entre MQTT-SN y MQTT requiere del entendimiento de las estructuras de mensajes tanto de MQTT como MQTT-SN, estructuras que se las puede encontrar en las siguientes referencias (Stanford-Clark & Truong, 2013) (MQTT, 2022) y que no se detallan en el presente artículo.

El proceso inicia con un mensaje CONNECT enviado por el cliente o nodo sensor hacia el Gateway, en la Figura 7 se muestra la estructura del mensaje enviado.

**Figura 7***Mensaje CONNECT*

Length (byte 0)	MsgType (1)	Flags (2)	ProtocolID (3)	Duration (4,5)	ClientID (6:n)
--------------------	----------------	--------------	-------------------	-------------------	-------------------

El Gateway recibe el mensaje, detecta que se trata de una nueva conexión, analiza las banderas del campo Flags y procede a enviar el mensaje WILLTOPICREQ para solicitar al cliente que envíe su WillTopic, la estructura del mensaje enviado se observa en la Figura 8.

**Figura 8***Mensaje WILLTOPICREQ*

Length (byte 0)	MsgType (1)
--------------------	----------------

Posteriormente del lado del cliente se envía el mensaje WILLTOPIC, el cual tiene la estructura de la Figura 9.

**Figura 9***Mensaje WILLTOPIC*

Length (byte 0)	MsgType (1)	Flags (2)	WillTopic (3:n)
--------------------	----------------	--------------	--------------------

Una vez recibido el mensaje WILLTOPIC, el Gateway envía el mensaje WILLMSGREQ, con la estructura de la Figura 10 pero con el campo MsgType en 0x08. Con este mensaje se solicita al nodo cliente que envíe su información en el campo WillMsg.

**Figura 10***Mensaje WILLMSGREQ*

Length (byte 0)	MsgType (1)	WillMsg (2:n)
--------------------	----------------	------------------

Como respuesta a este último mensaje, el Gateway envía un mensaje CONNACK para indicar que el proceso de establecimiento de conexión está completo y el nodo cliente ya puede comunicarse con el Broker MQTT. La estructura y campos del mensaje CONNACK se muestra en la Figura 11.

**Figura 11***Mensaje CONNACK*

Length (byte 0)	MsgType (1)	ReturnCode (2)
--------------------	----------------	-------------------

En el diagrama de estados propuesto, al culminar el proceso del estado ESTABLECIMIENTO se realiza una transición hacia el estado DISPONIBLE para que el convertidor de protocolos pueda continuar con su operación.

**Estado REGISTRO**

Después de que el nodo cliente haya establecido una conexión primaria con el Gateway, ya está en capacidad de comunicarse con el Bróker MQTT, pero antes debe realizar un proceso



de registro en el presente estado. Debido al limitado ancho de banda de las redes inalámbricas de sensores, no se puede enviar la información junto con el nombre del tópico como se hace en MQTT. Es por esto que en el protocolo MQTT-SN se desarrolló el registro denominado TopicName, en el cual se reemplaza el nombre de tópico por un identificador más corto denominado TopicId. Para iniciar el registro, el cliente envía un mensaje REGISTER al Gateway, con la estructura de la Figura 12.

**Figura 12**

*Mensaje REGISTER*

Length (byte 0)	MsgType (1)	TopicID (2,3)	MsgID (4,5)	TopicName (6:n)
--------------------	----------------	------------------	----------------	--------------------

Como respuesta al mensaje REGISTER, el Gateway genera el mensaje REGACK, Figura 13, el cual tiene la siguiente estructura.

**Figura 13**

*Mensaje REGACK*

Length (byte 0)	MsgType (1)	TopicID (2,3)	MsgID (4,5)	ReturnCode (6)
--------------------	----------------	------------------	----------------	-------------------

Una vez culminado este proceso, el nodo sensor ya está apto para publicar información en el Bróker MQTT con el objetivo de que lo almacene o reproduzca según el requerimiento de la aplicación. A continuación, luego realizar los procesos en los estados ESTABLECIMIENTO y REGISTRO, se realiza la conversión de protocolos. Luego de finalizar el registro del nodo cliente, se realiza la transición al estado DISPONIBLE y se espera el arribo de mensajes dirigidos hacia el Bróker.

### Estado IDENTIFICACION

Si el nodo cliente de la red inalámbrica de sensores, ya realizó su conexión primaria con el Gateway y también registro su nombre del tópico, ya puede comunicarse con el Broker, el cual está encargado de la adaptación de los mensajes entre los protocolos MQTT-SN y MQTT. El momento que en el estado RECEPCION se determina que el mensaje recibido en el Gateway va dirigido al Bróker, se realiza la transición al estado de IDENTIFICACIÓN.

En este estado se analiza la cabecera del mensaje recibido, específicamente el campo MsgType, el cual contiene un código de identificación que corresponde a alguno de los mensajes definidos en el protocolo MQTT-SN. Después de analizar el campo MsgType se identifica el mensaje recibido y se procede inmediatamente a realizar una transición al estado EXTRACCIÓN.

### Estado EXTRACCIÓN

Una vez concluido el proceso de identificación del mensaje MQTT-SN, en el estado EXTRACCIÓN se analiza la salida del estado IDENTIFICACIÓN para conocer el mensaje MQTT-SN transferido al Gateway. Se emplea campos del mensaje que serán usados para la creación del mensaje MQTT en el estado REESTRUCTURACIÓN.

### Estado REESTRUCTURACIÓN

En este estado se reciben los campos del mensaje MQTT-SN para realizar la adaptación al mensaje MQTT y ser enviados al bróker. En el estado REESTRUCTURACIÓN el procedimiento que se realiza es una asignación de un campo MQTT-SN al correspondiente

campo MQTT, para esto en algunos casos se requiere información previa del nodo sensor que esta almacenada en el Gateway, gracias a la comunicación previa que se realiza con el Bróker.

Una vez finalizado el proceso de adaptación y generación del mensaje MQTT realizado en el presente estado, se realiza el envío del mensaje hacia el Bróker, al terminar este evento se produce una transición hacia el estado base DISPONIBLE. Es importante aclarar que la conversión realizada en el Gateway depende del mensaje de control enviado por el nodo sensor.

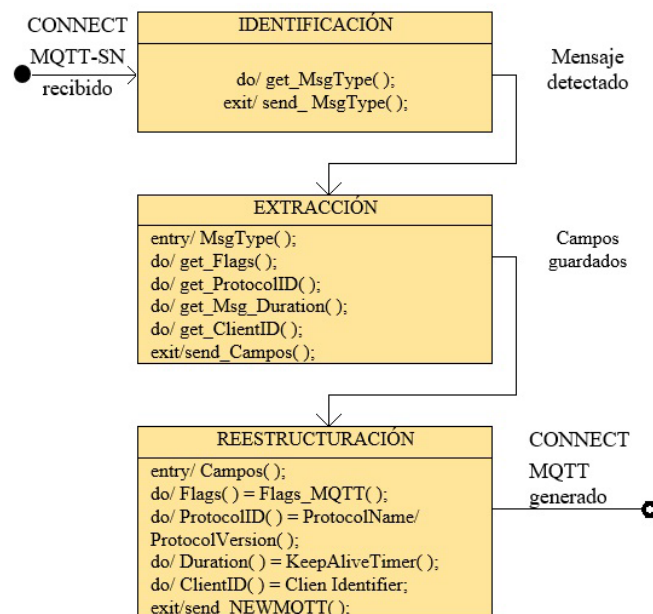
A continuación, se presentan los procesos de conversión realizados en los últimos estados del diagrama para cada mensaje del protocolo MQTT-SN así como los campos del mensaje que requieren ser modificados.

## Mensaje CONNECT

En la Figura 14 se presenta los procesos realizados en el mensaje CONNECT.

**Figura 14**

*Conversión realizada en el mensaje CONNECT*



Los cambios del contenido de los campos del mensaje se presentan a continuación:

- Length y MsgType no sufren cambios.
- Flags:
  - No se usan DUP, QoS, Retain y TopicIdType.
  - Will: si esta activa, precisa que el cliente solicita el indicador de tópico Willtopic y un indicador de mensaje WillMessage.
  - CleanSession: Al igual que en MQTT, si está activo se borra la sesión.
- ProtocolId: el Broker realiza una correlación correspondiente a su ProtocoloName y ProtocolVersion.
- Duration: Contiene el valor de KeepAliveTimer.
- ClientId: El Broker hace la correlación correspondiente a ClientIdentifier.

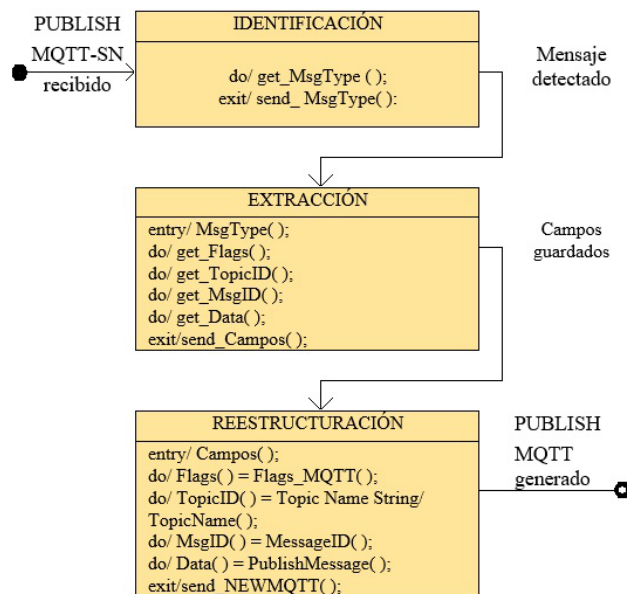
## Mensaje PUBLISH

En la Figura 15 se presenta los procesos realizados en el mensaje PUBLISH. Los cambios del contenido de los campos del mensaje se presentan a continuación:

- Length y MsgType no sufren cambios.
- Flags:
  - DUP, QoS y Retain no sufren cambios en el protocolo MQTT.
  - Will y CleanSession no se usan.
- TopicId: El Broker realiza una correlación a su correspondiente TopicName.
- MsgId: Corresponde a MessageID en el protocolo MQTT.
- Data: Contiene la información publicada.

**Figura 15**

*Conversión Realizada en el mensaje PUBLISH*

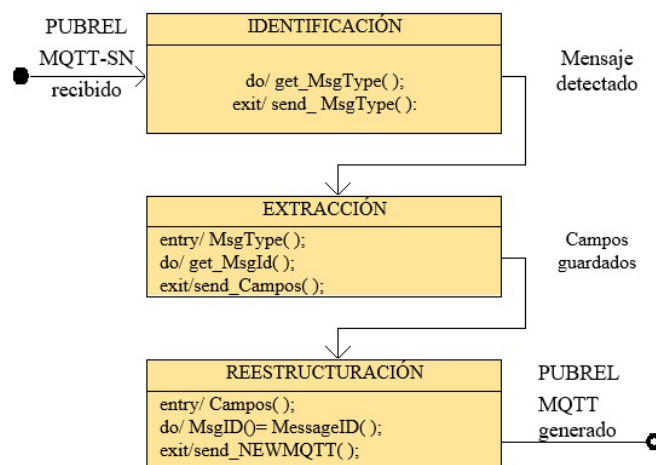


## Mensaje PUBREL

En la Figura 16 se presenta los procesos realizados en el mensaje PUBREL.

**Figura 16**

*Conversión realizada en el mensaje PUBREL*

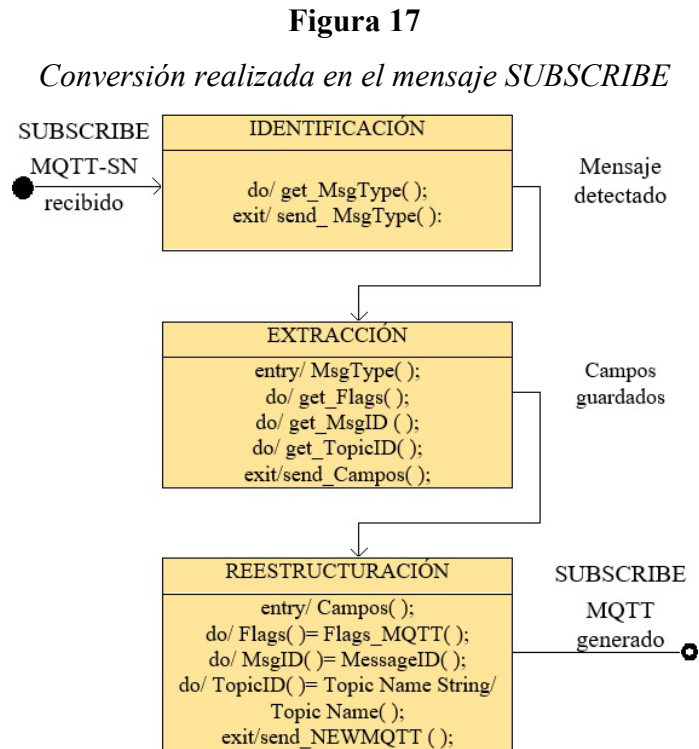


Los cambios del contenido de los campos del mensaje se presentan a continuación:

- Length y MsgType no sufren cambios.
- MsgId: Es el mismo valor del mensaje PUBLISH.

### Mensaje SUBSCRIBE

En la Figura 17 se presenta los procesos realizados en el mensaje SUBSCRIBE.



Los cambios del contenido de los campos del mensaje se presentan a continuación:

- Length y MsgType no sufren cambios.
- Flags:
  - DUP y QoS no sufren cambios en el protocolo MQTT.
  - Retain, TopicIdType, Will y CleanSession no se usan.
- MsgId: Corresponde al campo MessageID en el protocolo MQTT.
- TopicId: El Bróker realiza una correlación a su correspondiente TopicName.

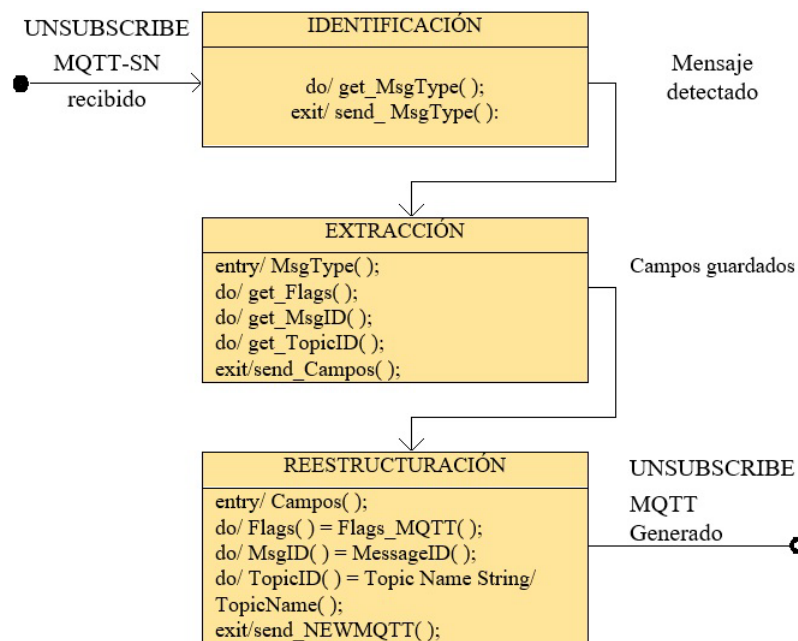
### Mensaje UNSUBSCRIBE

En la Figura 18 se presenta los procesos realizados en el mensaje UNSUBSCRIBE. Los cambios del contenido de los campos del mensaje se presentan a continuación:

- Length y MsgType no sufren cambios.
- Flags:
  - No se usan DUP, QoS, Retain, Will y CleanSession.
  - TopicIdType: Indica el tipo de identificador para con el Broker.
  - Clean Session: Al igual que en MQTT, si está activo se borra la sesión.
- MsgId: Corresponde al campo MessageID en el protocolo MQTT.
- TopicId: El Broker realiza una correlación a su correspondiente TopicName.

**Figura 18**

*Conversión realizada en el mensaje UNSUBSCRIBE*

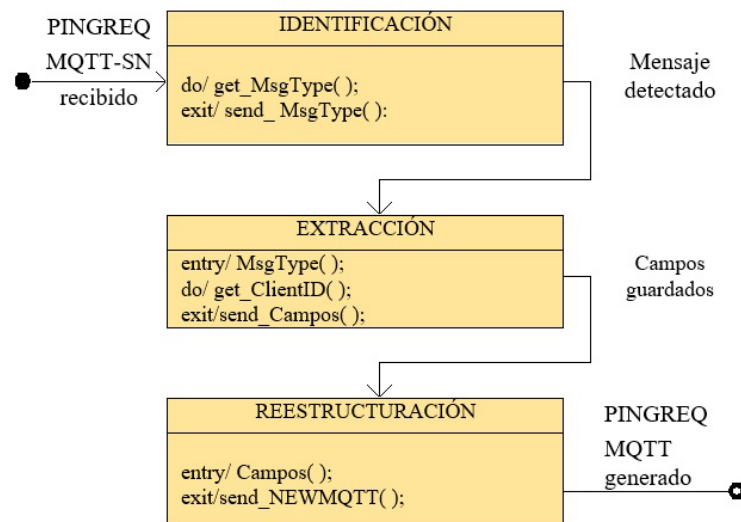


### Mensaje PINGREQ

En la Figura 19 se presenta los procesos realizados en el mensaje PINGREQ.

**Figura 19**

*Conversión realizada en el mensaje PINGREQ*



Los cambios del contenido de los campos del mensaje se presentan a continuación:

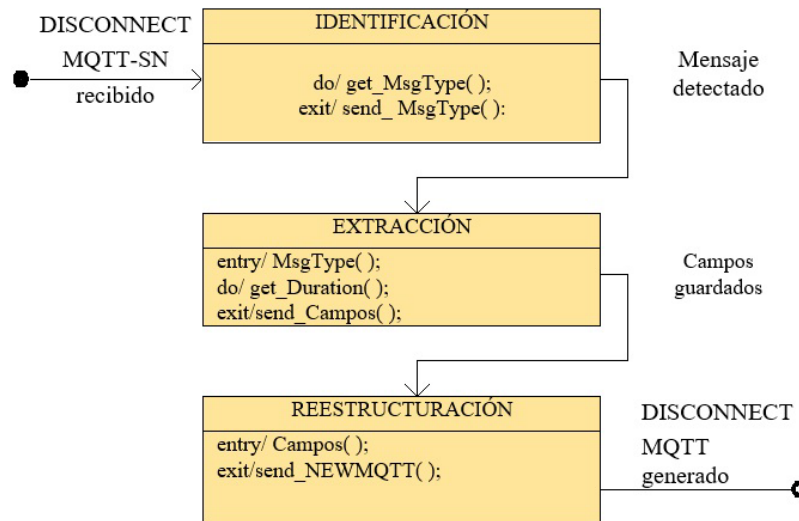
- Length y MsgType no sufren cambios.
- ClientId: Campo opcional, que se incluye el momento que un cliente que está en modo reposo pasa al estado activo para esperar los mensajes enviados por el Bróker/Gateway.

### Mensaje DISCONNECT

En la Figura 20 se presenta los procesos realizados en el mensaje DISCONNECT.

**Figura 20**

*Conversión realizada en el mensaje DISCONNECT*

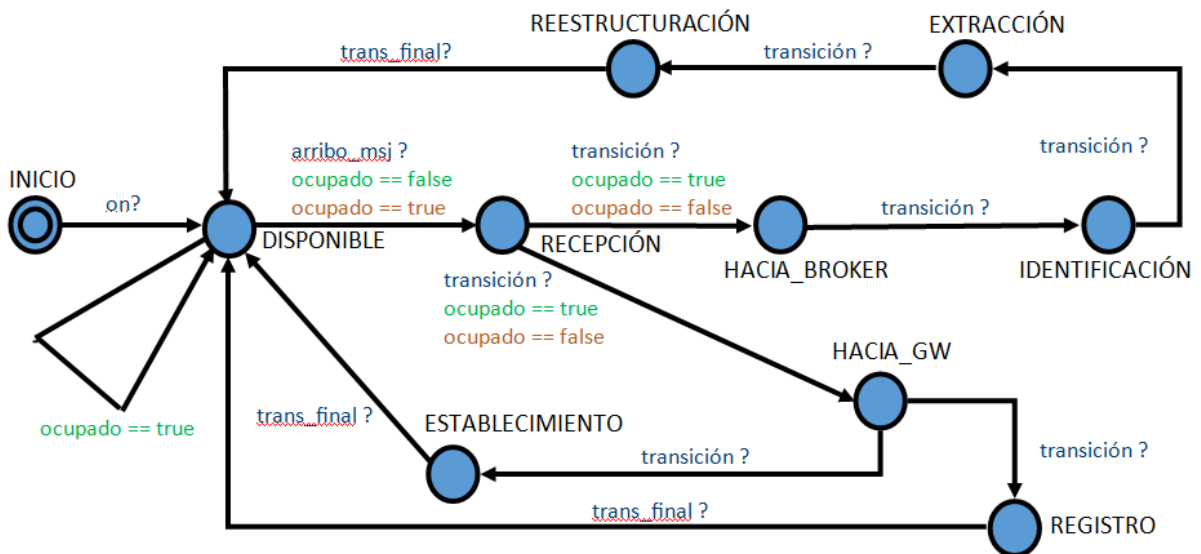


En el mensaje enviado por un cliente que desea cerrar la conexión, el campo opcional *Duration* se utiliza si el cliente pasa a estado de reposo. El mensaje no sufre cambios en MQTT.

Finalmente, para encontrar inconsistencias o solucionar problemas de lógica en los estados y procesos definidos, se utilizó UPPAAL, el cual es un entorno de herramientas que se utilizan para modelar, simular y verificar sistemas de tiempo real, desarrollado por las universidades de Uppsala y de Aalborg (Uppsala University, 2019). En la Figura 21 se presenta el diagrama de estados del convertidor de protocolos MQTT-SN a MQTT desarrollado en UPPAL.

**Figura 21**

*Diagrama de estados desarrollado en UPPAL*



## Conclusiones

Se analizó la base teórica y operativa de los protocolos MQTT y MQTT-SN, las estructuras de los mensajes, para plantear el diagrama de estados para la convertibilidad de estos protocolos. Los diagramas de estado obtenidos, tienen el propósito de ser una referencia para la futura implementación del convertidor de protocolos en un nodo Gateway para permitir la conectividad entre nodos inalámbricos de sensores que operan con MQTT-SN y el servidor MQTT.

La conversión del protocolo MQTT-SN a MQTT requiere de procesos previos para establecer un canal de comunicación entre el nodo sensor y el nodo Gateway debido a las limitaciones en la capacidad de transmisión del protocolo MQTT-SN, por lo que hay mensajes específicos definidos en este protocolo que no tienen un equivalente en el protocolo MQTT y cuya información debe ser adquirida previamente por el Gateway que va a realizar la conversión. Se realizó el análisis de los campos presentes en los mensajes definidos en cada protocolo para poder establecer la correspondencia de datos en el proceso de conversión.

El diagrama de estados diseñado en el presente trabajo representa el funcionamiento del convertidor de protocolos, y los procesos realizados en cada estado, los eventos que originan las diferentes transiciones y los resultados obtenidos a la salida.

El diagrama de estados desarrollado en UML, es una guía para futuras implementaciones del convertidor de protocolos con cualquier lenguaje de programación orientado a objetos y contribuir al desarrollo de una arquitectura para redes inalámbricas de sensores con topología lineal a gran escala.

Bajo este contexto, el diagrama de estados permite realizar a futuro la codificación e implementación del convertidor de los protocolos, en nodos con diferente hardware.

## Referencias

- Acosta, C. E., Gil-Castiñeira, F., Costa-Montenegro, E., & Silva, J. S. (2021). Reliable link level routing algorithm in pipeline monitoring using implicit acknowledgements. *Sensors (Switzerland)*, 21(3). <https://doi.org/10.3390/s21030968>
- Adepoju, O. (2022). Internet of Things (IoT). In *Springer Tracts in Civil Engineering*. [https://doi.org/10.1007/978-3-030-85973-2\\_8](https://doi.org/10.1007/978-3-030-85973-2_8)
- Al Rasyid, M. U. H., Astika, F., & Fikri, F. (2019). Implementation MQTT-SN Protocol on Smart City Application based Wireless Sensor Network. *Proceeding - 2019 5th International Conference on Science in Information Technology: Embracing Industry 4.0: Towards Innovation in Cyber Physical System, ICSITech 2019*. <https://doi.org/10.1109/ICSITech46713.2019.8987546>
- Baquer, N. K., Al-modaffer, A. M., & Shahtoor, G. H. (2018). Throughput Study of IEEE 802 . 15 . 4 ZigBee-Based WSNs for Greenhouse Environments. *International Journal of Scientific Research Engineering & Technology (IJSRET)*, 7(3).
- Chen, Y., Hou, K. M., Zhou, H., Shi, H. L., Liu, X., Diao, X., Ding, H., Li, J. J., & De Vaulx, C. (2011). 6LoWPAN stacks: A survey. *7th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM 2011*. <https://doi.org/10.1109/wicom.2011.6040344>
- Egas, C., & Gil-Castiñeira, F. (2020). Revisión de requisitos, protocolos y desafíos en LWSN. *MASKAY*, 11(1). <https://doi.org/10.24133/maskay.v11i1.1728>
- Eghonghon Ukhurebor, K., Odesanya, I., Soo Tyokighir, S., George Kerry, R., Samson Olayinka, A., & Oluwafemi Bobadoye, A. (2021). Wireless Sensor Networks: Applications and Challenges. In *Wireless Sensor Networks - Design, Deployment and Applications*. <https://doi.org/10.5772/intechopen.93660>

- EMQX. (2020). *MQTT-SN protocol gateway* (4.3). [https://www.emqx.io/docs/en/v4.3/modules/mqtt\\_sn\\_protocol.html#protocol-introduction](https://www.emqx.io/docs/en/v4.3/modules/mqtt_sn_protocol.html#protocol-introduction)
- Fontes, F., Rocha, B., & Mota, A. (2020). Extending MQTT-SN with Real-Time Communication Services. *25th IEEE International Conference on Emerging Technologies and Factory Automation*, 1–4. <https://doi.org/10.1109/ETFA46521.2020.9212147>
- Koc, H., Erdogan, A., & Barjakly, Y. (2021). UML Diagrams in Software Engineering Research: A Systematic Literature Review. *The 7th International Management Information Systems Conference*. <https://doi.org/10.3390/proceedings2021074013>
- Kumar S., A. A., Ovsthus, K., & Kristensen., L. M. (2014). An industrial perspective on wireless sensor networks-a survey of requirements, protocols, and challenges. *IEEE Communications Surveys and Tutorials*, 16(3), 1391–1412. <https://doi.org/10.1109/SURV.2014.012114.00058>
- Luo, F., Feng, T., & Zheng, L. (2021). Formal Security Evaluation and Improvement of Wireless HART Protocol in Industrial Wireless Network. *Security and Communication Networks*, 2021. <https://doi.org/10.1155/2021/8090547>
- MQTT. (2022). *MQTT Specifications*. <https://mqtt.org/mqtt-specification/>
- Mura, M., & Sami, M. G. (2008). Code generation from statecharts: Simulation of wireless sensor networks. *Proceedings - 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, DSD 2008*. <https://doi.org/10.1109/DSD.2008.106>
- Paho. (2021). *MQTT-SN Transparent Gateway*. <https://www.eclipse.org/paho/index.php?page=components/mqtt-sn-transparent-gateway/index.php>
- Park, C.-S., & Nam, H.-M. (2020). Security Architecture and Protocols for Secure MQTT-SN. *IEEE Access*, 8, 226422–226436. <https://doi.org/10.1109/ACCESS.2020.3045441>
- Queiroz, D. V., Alencar, M. S., Gomes, R. D., Fonseca, I. E., & Benavente-Peces, C. (2017). Survey and systematic mapping of industrial Wireless Sensor Networks. In *Journal of Network and Computer Applications* (Vol. 97). <https://doi.org/10.1016/j.jnca.2017.08.019>
- Raptis, T. P., Passarella, A., & Conti, M. (2020). A survey on industrial internet with ISA100 wireless. *IEEE Access*, 8. <https://doi.org/10.1109/ACCESS.2020.3019665>
- Serianni, A., & De Rango, F. (2022). Application Layer Protocols for Internet of Things. In *Lecture Notes in Networks and Systems* (Vol. 289). [https://doi.org/10.1007/978-3-030-87049-2\\_18](https://doi.org/10.1007/978-3-030-87049-2_18)
- Stanford-Clark, A., & Truong, H. L. (2013). MQTT for sensor networks ( MQTT-SN). *Protocol Specification 1.2*.
- Uppsala University, A. U. (2019). *Uppaal* (4.1.25). <https://uppaal.org/>