

## SISTEMA DE SEÑALIZACIÓN DIGITAL SOPORTADO EN ECOSISTEMA IOT

Guillermo Lastre Olazábal<sup>1</sup>, Gilbert Rafael García Cabrera<sup>2</sup>

<sup>1</sup> CITMATEL. [laestre@citmatel.cu](mailto:laestre@citmatel.cu), <sup>2</sup> CITMATEL. [gilbert@citmatel.cu](mailto:gilbert@citmatel.cu)

### RESUMEN

En este trabajo se implementa una aplicación de Cartelería Digital que a nivel de hardware y software permite el control centralizado, dinámico y seguro de múltiples pantallas en una interfaz única. Posee la capacidad de ser insertado en un ecosistema IoT a través del protocolo MQTT, lo que le permite reaccionar ante determinadas situaciones detectadas por sensores u otros dispositivos, variando de manera dinámica el contenido que muestra en las pantallas. Se despliegan, además, las plataformas de IoT que permiten la administración y monitoreo de las pantallas, tanto dentro de la red local (*edge*) como basadas en nube. Se demuestra su funcionamiento con un caso de estudio donde la pantalla cambia la imagen mostrada cuando un sensor de movimiento detecta personas, además de encender las luces del local que permanecían apagadas mientras que no hubiera personas en él. De esta forma se hace un uso eficiente de la energía, ya que solo se activa cuando es necesario.

**PALABRAS CLAVES:** Cartelería Digital, Internet de las Cosas, Protocolo MQTT.

### DIGITAL SIGNALING SYSTEM FOR IOT ECOSYSTEMS

### ABSTRACT

In this work, a Digital Signaling application is implemented that, at the hardware and software level, allows centralized, dynamic, and secure control of multiple screens in a single interface. It can be inserted into an IoT ecosystem through the MQTT protocol, which allows it to react to certain situations detected by sensors or other devices, dynamically changing the content displayed on the screens. In addition, IoT platforms are deployed that allow the management and monitoring of the screens, both within the local network (*edge*) and based on the cloud. Its operation is demonstrated with a case study where the screen changes the image shown when a motion sensor detects people and turns on the lights of the premises that remain off. At the same time, there were no people in it. This way, energy is used efficiently since it is only activated when necessary.

**INDEX TERMS:** Digital Signage, Internet of Things (IoT), MQTT Protocol.

## 1. INTRODUCCIÓN

La cartelería digital (*digital signage* en idioma inglés), conocida también como señalización digital dinámica, señalización digital multimedia o cartelería digital, es el uso de contenidos digitales emitidos a través de pantallas (monitores LCD, pantallas de TV, paneles de LED o proyectores), con fines informativos y/o publicitario [1]. Las pantallas que se emplean en estos sistemas están conectadas entre sí por una red local y administrada centralmente por una aplicación que gestiona su funcionamiento y la planificación del contenido a mostrar. Estos sistemas son extremadamente caros y requieren el uso de licencias para utilizar la herramienta de control centralizado, la cual suele ser propia del fabricante de las pantallas por lo que una aplicación desarrollada dentro de nuestro país se vuelve una alternativa necesaria.

En Cuba, a partir de la informatización de la sociedad se comienza a explotar incipientemente las bondades de este tipo de sistema. Existen ejemplos de su uso a nivel institucional y donde más se puede ver su empleo de una manera muy rudimentaria es en negocios particulares. Un ejemplo de este sistema empleado en Cuba a nivel estatal lo constituyen pantallas dedicadas al enfrentamiento al cambio climático, ubicadas en cuatro instituciones del CITMA y en la Universidad de La Habana. Estas pantallas

son fabricadas por SAMSUNG y son gestionadas de manera remota por el software MagicInfo [2]. Se ha identificado además el uso de este tipo de sistema por la Oficina del Historiador en la zona del casco histórico de la Habana y en algunas instituciones como museos y hoteles.

En este trabajo se describe el diseño e implementación de un sistema de Cartelería Digital basado en hardware de bajo costo y software libre. Estos componentes tienen la limitante de que sólo se permite el control efectivo de una sola pantalla. La solución que se presenta propone el diseño, desarrollo e implementación de una herramienta informática que permita el control centralizado y seguro de múltiples clientes de Screenly OSE, en una interfaz única y con capacidad de manipular de manera dinámica y reactiva su contenido, mediante su inserción en un Ecosistema de IoT. Por último, se presenta un caso de estudio donde se valida la funcionalidad de la solución propuesta.

## **2. DESCRIPCIÓN DEL SISTEMA DE SEÑALIZACIÓN DIGITAL IMPLEMENTADO.**

El diseño del sistema de señalización digital propio está basado en software y hardware libre y además, interactúa en un ecosistema IoT con otros dispositivos. A continuación, se describen brevemente algunos conceptos importantes para comprender el alcance de la solución propuesta.

### **Señalización digital**

Las soluciones de señalización digital actuales ofrecen una amplia gama de funciones y oportunidades de presentación, que incluyen video, audio, imágenes fijas y HTML. Puede mostrar el mismo contenido o diferente en varias pantallas. También es posible programar el contenido para que cambie a intervalos u horarios regulares, en función de mensajes distintos o audiencias cambiantes.

### **Componentes de un sistema de señalización digital**

Un sistema de señalización digital requiere para su funcionamiento una combinación de hardware específico para reproducir el contenido e interactuar con la pantalla conocido como reproductor o *Player*, un software especial para el reproductor de los medios digitales y la gestión de contenidos, una herramienta de control centralizado y tecnología que permita la actualización remota, como redes de datos locales o conectividad con Internet.

### **Características imprescindibles de un sistema de señalización digital**

Existen diversas opciones cuando se trata de software de señalización digital. Para elegir el mejor software de señalización digital es necesario tener en cuenta las siguientes características: Múltiples formatos y facilidad de programación de la visualización del mismo, soporte de múltiples pantallas, acceso multiusuario, supervisión e informes de red, Por último, implementar medidas de seguridad razonable que minimicen las vulnerabilidades que pueden tener este tipo de sistemas.

### **Aplicaciones de señalización digital propietarias, gratuitas y de código abierto**

En la actualidad existen variados software y sistemas propietarios [2] [3] que proporcionan soluciones de Cartelería Digital. Muchos de estos están constituidos por equipos físicos y servidores dedicados o están basados en la nube. El costo de estos sistemas suele ser elevado y además necesitan ser importados del extranjero. El bloqueo económico financiero de los Estados Unidos hacia Cuba hace difícil obtener los componentes necesarios para este tipo de sistema en nuestro país. La política de migración al software libre de Cuba, hace que la alternativa de una aplicación de pago no sea la más adecuada.

De igual forma existen alternativas gratuitas y de código abierto [4] las cuales fueron analizadas como posible solución, seleccionándose como aplicación para el reproductor de medios, la propuesta de Screenly OSE [5] como la más adecuada a nuestros requisitos. Las soluciones de señalización digital mencionadas anteriormente no tienen la funcionalidad deseada de tener un control centralizado ni de

interactuar con un ecosistema IoT, por lo que se hizo necesario la modificación de la aplicación seleccionada para el reproductor y la confección de una aplicación de control propia.

## Ecosistema IoT

Internet de las cosas (IoT, Internet of Things) es un concepto y un paradigma que considera la presencia generalizada en el entorno de una variedad de cosas/objetos que a través de conexiones inalámbricas o cableadas y esquemas de direccionamiento únicos pueden interactuar entre sí y cooperar con otras cosas/objetos para crear nuevas aplicaciones/servicios y alcanzar objetivos comunes.

Un Ecosistema IoT está basado en lo fundamental en una gran cantidad de dispositivos embebidos de bajo costo, escasa capacidad de cálculo y recursos, con un gran número de comunicaciones simultáneas, lo que define que el protocolo de comunicación IoT tiene que tener poca capacidad de procesamiento. Son sistemas escalables e interoperables (variedad de dispositivos, sistemas operativos y lenguajes de programación). La seguridad es un aspecto importante que muchas veces no se tiene en cuenta.

## Protocolo MQTT

MQTT (*MQ Telemetry Transport*) es uno de los protocolos de IoT más usados a nivel mundial debido a su flexibilidad y altas capacidades de comunicación, además de ser un software de estándar abierto. Aunque inicialmente era un formato propietario, en 2010 fue liberado y pasó a ser un estándar en 2014 según la OASIS (*Organization for the Advancement of Structured Information Standards*) [6]. MQTT es extremadamente simple y liviano, diseñado para dispositivos restringidos y redes de bajo ancho de banda, alta latencia o poco confiables. Sus principios de diseño están destinados a minimizar el ancho de banda de la red y los requisitos de recursos del dispositivo al mismo tiempo que se intenta garantizar la confiabilidad y cierto grado de garantía de entrega. El funcionamiento de este protocolo es un servicio de mensajería con patrón publicador/suscriptor (pub-sub) en el cual los clientes se conectan con un servidor central denominado *broker*.

## Estructura general y diseño del sistema implementado

Teniendo en cuenta los componentes clásicos de un sistema de este tipo, la solución propuesta (ver Fig. 1) emplea como reproductor de medios una minicomputadora Raspberry Pi 3b+ y la aplicación de código abierto Screenly OSE, interconectados con un Sistema de Control Centralizado, de manera inalámbrica o cableada en una red local propia,

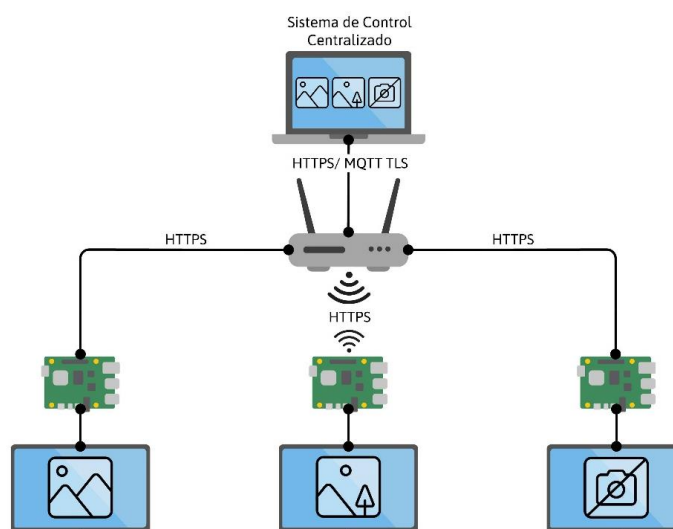


Figura 1: Esquema general de la solución propuesta.

**Unidad de visualización:** Puede ser cualquier tipo de pantalla que posea entrada HDMI: Monitor de PC, pantalla publicitaria, TV Wall o similar.

**Reproductor:** Basado en una Raspberry Pi 3b+, ejecuta el software, que permite la visualización, programación y monitoreo de la información a mostrar, se conecta a la pantalla directamente por HDMI y a la red por cable Ethernet o WiFi. Permite, además, el control del encendido de la pantalla.

**Aplicación de control centralizado:** Gestiona y monitoriza las pantallas para diferentes proyectos (locaciones), comunicándose directamente con el reproductor a través de HTTPS y con las plataformas IoT a través del protocolo MQTT, también de manera segura.

**Conectividad:** Los reproductores se pueden conectar a la red local usando la red cableada o de manera inalámbrica.

**Plataformas IoT:** Este es un componente opcional. Permiten la interacción del sistema de cartelería con dispositivos IoT para la variación dinámica del contenido a través de protocolos propios de este tipo de tecnología. Puede interactuar con plataformas IoT dentro del entorno de la red local (*edge computing*) o a nivel de nube (*cloud computing*) en dependencia del tipo de acceso que requiera el cliente final del servicio.

Este sistema tiene asociado un servicio integral que provee la empresa CITMATEL, que comprende: identificar las necesidades o requisitos del cliente; elaboración de tarea técnica; suministro de los materiales, insumos y equipos necesarios; el cableado, instalación y configuración de equipos; instalación de la solución centralizada; puesta a punto del sistema y entrenamiento al personal. Opcionalmente si el cliente lo solicita se asume el diseño, actualización, monitoreo y mantenimiento de la información de la campaña publicitaria.

## Características y funcionalidades del Sistema.

- Visualización de contenido web y multimedia (imágenes y videos) en múltiples pantallas.
- Creación de proyectos independientes (instituciones o locaciones) que agrupen múltiples clientes (pantallas) y categorías (clasificación de la información).
- Ejercer acciones de control sobre el hardware de visualización: Apagar o reiniciar el reproductor, apagar o encender las pantallas.
- Control centralizado de todos los proyectos y clientes desde una sola interfaz web.
- Mostrar en tiempo real el contenido de cada pantalla en la interfaz web de la aplicación.
- Planificación centralizada del contenido a mostrar en cada pantalla de manera independiente o grupo de pantallas seleccionado.
- Control de acceso a la aplicación de control centralizado y a los reproductores mediante usuario y contraseña.
- Comunicación segura (TLS) entre los diferentes elementos del sistema y autenticación de usuarios.

- Solución basada en software libre, cuyas licencias permiten la modificación y comercialización del resultado obtenido.
- Funciona de manera nativa sobre Raspberry Pi que permite emplear hardware de bajo costo como reproductor, permitiendo además otras posibilidades como el control de sensores o actuadores sobre la misma pantalla.

La solución propuesta inserta al sistema de señalización digital en un ecosistema IoT, la interrelación con sistemas inteligentes permitiría variar la información a mostrar de manera reactiva, por ejemplo: La conexión de sensores de presencia permite activar el encendido de la pantalla; la presencia de botones físicos o virtuales permitirían la selección del contenido a visualizar; la conexión de sensores ambientales permitiría mostrar datos actualizados de variables meteorológicas locales o de otra índole; la conexión a sistemas de vigilancia mediante cámaras o sistemas de control de acceso permitiría su uso en la seguridad física de las instalaciones.

## Descripción del software

La aplicación diseñada e implementada es un software intuitivo y fácil de usar por el personal al que está dirigido y lo suficientemente flexible para agregar nuevas funcionalidades en el futuro. A continuación, se brinda una descripción de algunas de las prestaciones y herramientas que se implementaron en la solución. Se define como Proyecto al conjunto de pantallas (Clientes) que conforman el sistema de señalización digital de una locación específica. La solución puede soportar múltiples proyectos. Para crear un Cliente, dentro de un proyecto se debe introducir un nombre único el cual será su identificador y una dirección IPv4 única, correspondiente a la dirección actual del reproductor en la red local. Una vez definidos los clientes y su autenticación, el usuario será capaz de saber en tiempo real si estos están conectados, desconectados o ver la imagen actual que se muestra en la pantalla (ver Fig. 2). Un usuario autenticado será capaz de gestionar, mediante la interfaz de la aplicación, el contenido a mostrar en cada Cliente, así como su planificación (ver Fig. 3).

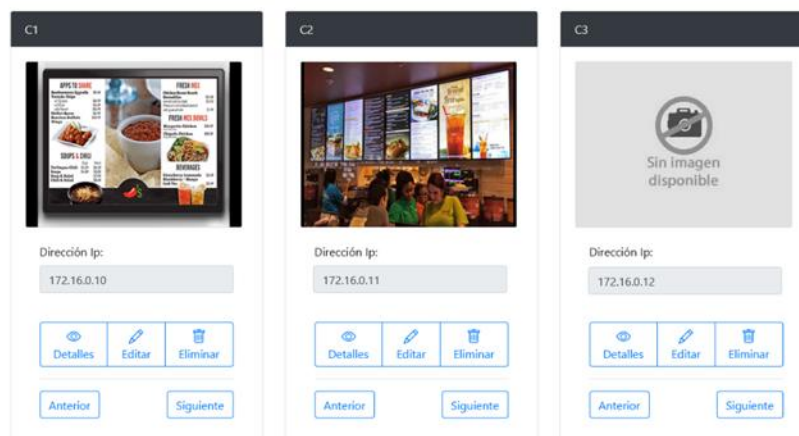


Figura 2: Vista de los clientes de un proyecto.

Como solución a la funcionalidad de interactuar con un ecosistema IoT, se implementó una opción independiente con la capacidad de comportarse como un cliente MQTT. El cual es capaz de suscribirse a una plataforma IoT usando este tipo de protocolo y enlazar una o varias acciones al tópico al que se suscribe. Dichas acciones están relacionadas con encender o apagar el reproductor, mostrar la información precedente, la anterior o ir una específica, esta última pudiera no estar activa en la

programación de imágenes a visualizar. El protocolo MQTT trabaja directamente con el servidor de la plataforma IoT (tanto local como en la nube), basado en un patrón de publicación/subscription en un tópico o tema. En la Fig. 4 se muestra la pantalla de configuración de este protocolo en la aplicación, que incluye los datos del servidor *broker* al cual conectarse.

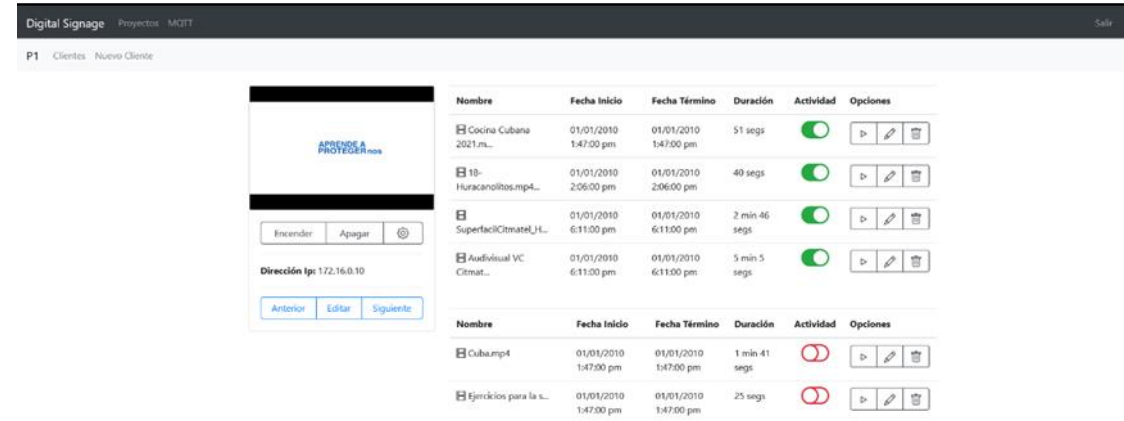


Figura 3: Vista de gestión de un cliente.

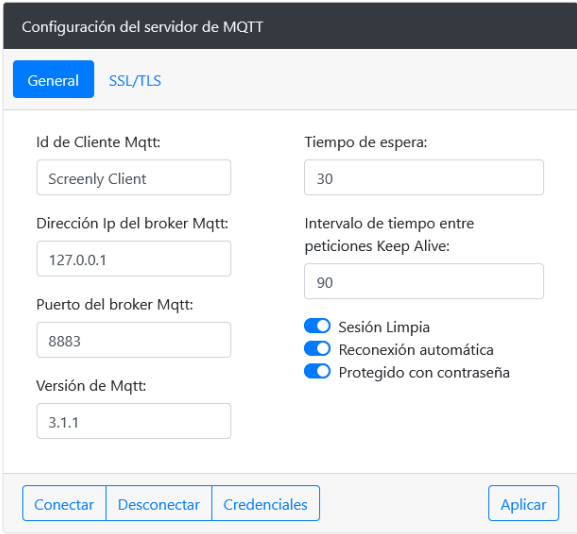


Figura 4: Configuración general del cliente MQTT.

En la Fig. 5, se muestran los detalles de cómo reaccionaría el sistema en dependencia del valor del tópico T1: el cliente C1 pasará a la siguiente información si el valor de T1 es mayor que cero, mientras que C2 y C3 visualizaran la imagen anterior, para este mismo valor del tópico.

**Seguridad implementada**

La seguridad implementada se basó en las capas de seguridad definidas para un ecosistema de IoT basado en el protocolo MQTT según se describe en [3]. La solución propuesta es la siguiente: Para la capa de seguridad a nivel de infraestructura de red, se propone que sea una red datos separada de la red de la institución, se toma en cuenta la seguridad que permite el equipo de conectividad utilizado: restricción a nivel de MAC de los equipos que se pueden conectar, uso de contraseñas fuertes en las conexiones WiFi, uso de VLANS, entre otras. Para la capa de seguridad a nivel de transporte se crean certificados auto

firmado para los servidores de las plataformas y la aplicación de control. En el caso de la solución de software en los reproductores (Screenly OSE) se habilitó la conexión segura de estos, siguiendo las instrucciones de su sitio web [5]. Esto permite que se implemente la encriptación SSL/TLS v1.2 a nivel de la capa de transporte también en el servidor *broker* [7]. Para la capa de seguridad a nivel de aplicación, se habilita en los reproductores, en el servidor IoT y en la aplicación de control la autenticación por nombre de usuario y contraseña. Al estar implementado la encriptación SSL/TLS, las credenciales de estos no viajan en texto plano. Con relación al SO Raspbian en el cual están basados los reproductores, se creó un nuevo usuario y contraseña para usuario y se implementaron otras medidas de seguridad [8]. Para esta solución no se implementó el cifrado de la carga útil o *payload* del tópico.

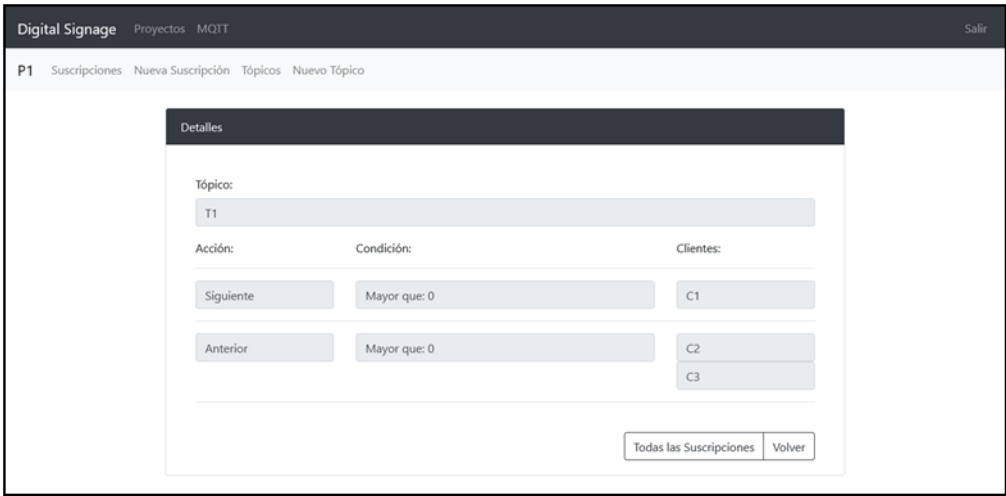


Figura 5: Vista de una suscripción.

**Descripción del Ecosistema IoT implementado.**

El sistema de cartelería digital propuesto se conecta a través del protocolo MQTT, con un ecosistema IoT implementado en el centro de datos de CITMATEL [9] [10], el cual está conformado a nivel de borde de la red (*edge computing*) por un servidor *broker* Mosquitto y por la plataforma IoT Thingsboard, basada en los principios de la computación en la nube (*cloud computing*). En la Fig. 6, se observa un esquema general de la solución propuesta.

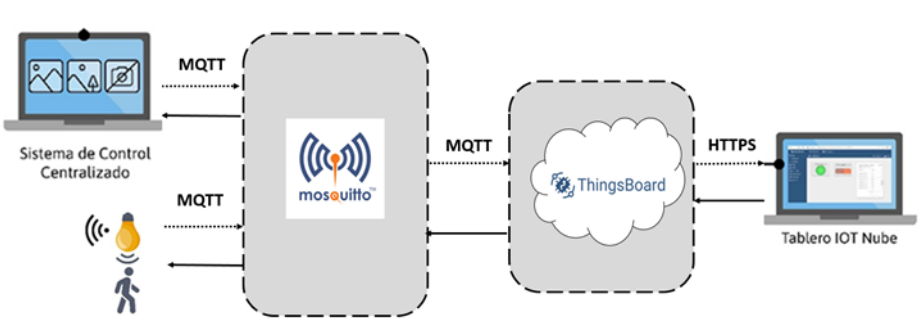


Figura 6: Esquema del Ecosistema IoT implementado.

Eclipse Mosquitto [11], [7] es un *broker* MQTT de código abierto (con licencia EPL/EDL), que implementa las versiones 5.0, 3.1.1 y 3.1 del protocolo MQTT. Mosquitto es ampliamente utilizado debido a su ligereza lo que permite, emplearlo fácilmente en gran número de ambientes, incluso si estos son de pocos recursos, es adecuado para usar en varios dispositivos, desde computadoras de una sola

placa de baja potencia, como una Raspberry Pi hasta servidores completos. Está disponible para una gran cantidad de plataformas de software: MacOS, Windows, y varias distribuciones de Linux. Admite la interconexión de varios servidores Mosquitto usando una propiedad conocida como *Bridge*. Permite implementar a nivel de la aplicación medidas de seguridad de alto impacto (autenticación, autorización y cifrado SSL/TLS). Tiene una gran comunidad asociada y es un sistema activo, la última versión 2.0.15 es de agosto del 2022.

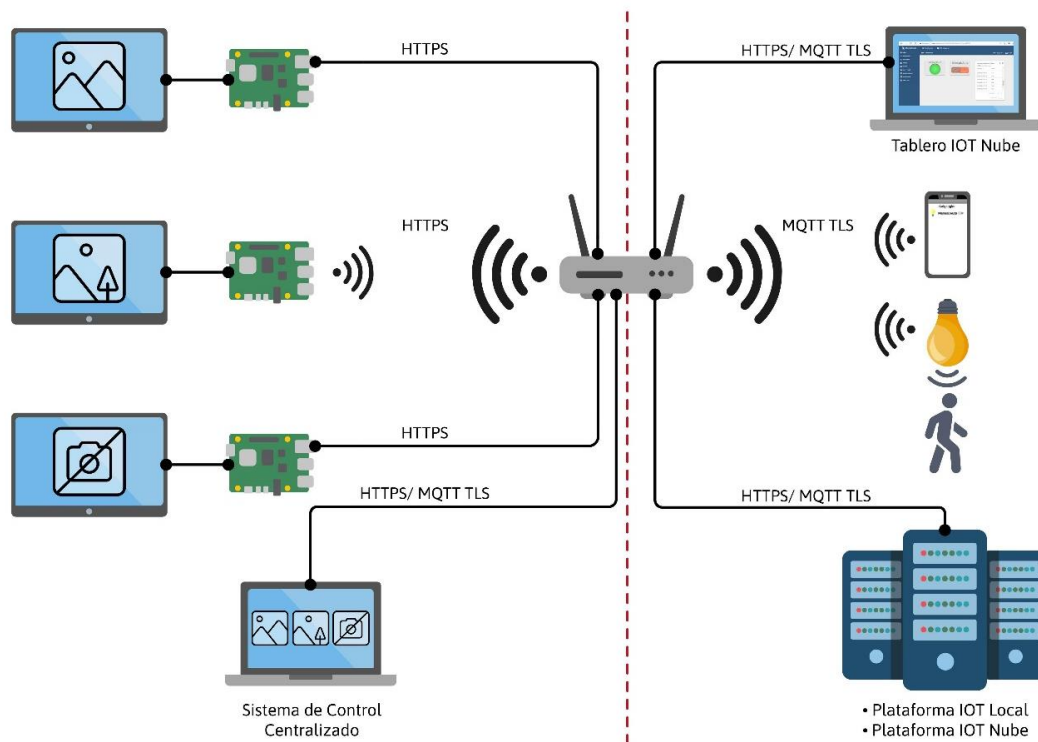


Figura 7: Esquema general del caso de estudio propuesto.

La plataforma IoT a nivel de Nube está implementada con un servidor ThingsBoard [12] Versión 3.0.0-1 de código abierto. Está centrada en permitir un rápido desarrollo, gestión y escalado de proyectos relacionados con esta tecnología. Permite la recopilación, el procesamiento, la visualización y la administración de dispositivos. Admite implementaciones locales y en la nube. Es compatible con los protocolos de IoT estándar de la industria: MQTT, CoAP y HTTP. Consigue combinar escalabilidad, tolerancia a los fallos y un buen rendimiento a la hora de capturar los datos del dispositivo para su procesamiento y control.

### 3. RESULTADOS

Como ejemplo de uso de la solución de cartelería digital se demuestra su funcionamiento con una aplicación práctica (Fig. 7) donde se pasa a la siguiente información de la pantalla del cliente C1, cuando un sensor de movimiento detecta personas. Esto sucede porque el sensor de movimiento publica en el tópico T1 un valor mayor que cero (ver Fig. 5), esto provoca además el encendido de las luces del local, las cuales permanecían apagadas mientras no hubiera personas en él [13]. Esta aplicación permite un uso eficiente de la energía ya que se pudiera configurar el encendido de las pantallas y las luces cuando detecta la presencia de personas.

La interacción del sistema de cartelería con la plataforma IoT a nivel de nube se logra a través de un tablero de mando implementado en el Thingsboard (Fig. 8), que permite mostrar los estados y los elementos de control con el empleo de sensores y actuadores de la lámpara inteligente. El tablero permite



interactuar con el servidor para obtener información e interactuar con ella. En este caso se configura el componente **Lámpara GL\_01** para ver cuando la lámpara está apagada o encendida.

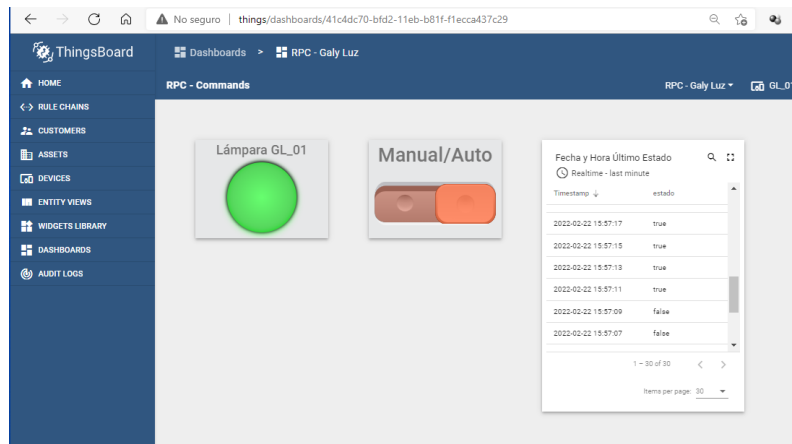


Figura 8: Tablero de mando implementado en ThingsBoard: Luz encendida, interruptor y tabla de últimas acciones.

## 4. CONCLUSIONES

Disponer de un Sistema de Señalización Digital propio permite adaptarse a cualquier cambio o inclusión de nuevas funcionalidades, disminuye costos de implementación y permite brindar servicios personalizados. Con el presente trabajo se logran establecer las bases de uno de los principales componentes de estos sistemas: el control centralizado, adaptado a las tendencias actuales y necesidades de las instituciones que necesiten emplearlos para sus implementaciones de publicidad o anuncios.

Como resultado del trabajo realizado se concluye que es posible la gestión centralizada de múltiples pantallas situadas en locaciones diferentes, permitiendo la planificación del contenido a mostrar, el monitoreo de su estado en tiempo real y la visualización al operador de la imagen que se está mostrando en la pantalla. Se demuestra que gracias al estar conectado a una plataforma IoT a través del protocolo MQTT, es posible reaccionar, a partir de percibir variaciones en los valores de los tópicos a los cuales está suscrito, con la ejecución de una acción definida por el usuario sobre uno o varios reproductores. Por último, se validó la solución en un caso de estudio en el que se obtuvieron los resultados esperados.

## 5. RECOMENDACIONES Y TRABAJO FUTURO.

A partir de la implementación práctica de este sistema surge la necesidad de incorporar otras funcionalidades no concebidas inicialmente, pero que son características de los Sistemas de Señalización Digital más avanzados. Con el objetivo de potenciar la seguridad y el control centralizado, se proponen los siguientes aspectos como trabajos futuros a realizar que permitirían incorporar las siguientes prestaciones al sistema: Creación de múltiples usuarios y autorización basada en roles para la aplicación, verificación por correo en la creación de los usuarios y para recuperar contraseña, mostrar imágenes en directo de una cámara web o una cámara IP e implementación de aplicación móvil propia para controlar la imagen en pantalla a partir de conectarse a la plataforma IoT.

## REFERENCIAS

- [1] ITU-H. *SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS. IPTV multimedia services and applications for IPTV – Digital Signage*, Recommendation ITU-T H.783. TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU. 2019

- [2] SAMSUNG. *MagicINFO™ 9* [en línea]. [ref. de 8 Agosto 2022]. Disponible en Web: <https://www.samsung.com/latin/business/display-solutions/magicinfo/>
- [3] GARCÍA, Gilber. Rafael; LASTRE, Guillermo. "Proyecto de desarrollo: Propuesta de implementación de un prototipo para servicio de cartelería digital empleando software libre". UEB proyectos de redes y Sistemas de cómputo. CITMATEL, La Habana, Nov 2022. 63 PP.
- [4] SoftwareSuggest. *14 Best Free & Open Source Digital Signage Software Mayo 2020*. [en línea]. [ref. de: 7 de Diciembre 2021]. Disponible en Web: <https://www.softwaresuggest.com/blog/best-free-open-source-digital-signage-software/>.
- [5] Screenly Inc. *Screenly Open Source Edition* [en línea]. [ref. de: 31 enero 2022]. Disponible en Web: <https://www.screenly.io/ose/>.
- [6] BANKS, Andrew; BRIGGS, Ed; BORGENDALE; Ken y GRUPA, Rahul. *MQTT Version 5.0. OASIS Standard. 07 Marzo 2019*. [en línea]. [ref. de: 8 de Agosto 2022]. Disponible en Web: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>.
- [7] COPE, Steve. *Steve's Internet Guide* [en línea]. [ref. de: 12 Agosto 2022]. Disponible en web: <http://www.steves-internet-guide.com/>.
- [8] LÓPEZ, Nacho. *Monitorización y Seguridad En La Raspberry Pi* [en línea]. [ref. de: 28 Julio 2022]. Disponible en Web: <https://nacholoop.github.io/enigma/seguridad.html#pass>.
- [9] LASTRE, Guillermo. "Proyecto de desarrollo SER 21119: Ecosistema IOT: Implementación de Servidor Broker IoT". UEB Proyectos de redes y Sistemas de Cómputo. CITMATEL, La Habana, Jul 2022. 50 PP.
- [10] GARCÍA, Roberto. "Proyecto de desarrollo SER 21120: Ecosistema IOT: Implementación de Servicios IoT en la Nube". UEB Proyectos de redes y Sistemas de cómputo. CITMATEL, La Habana, Jul 2022. 35 PP.
- [11] Eclipse Foundation. *Eclipse Mosquitto* [en línea]. [Ref. de: 12 Agosto 2022]. Disponible en Web: <https://mosquitto.org/>.
- [12] ThingsBoard Inc. *ThingsBoard 2022* [en línea]. [Ref de: 12 Agosto 2022]. Disponible en Web: <https://thingsboard.io/>.
- [13] GARCÍA, Roberto. "Proyecto de desarrollo SER 2199: Ecosistema IoT Implementación de Dispositivos y Sensores IOT (Diseño de dispositivos inteligentes Lámpara Inteligente)". UEB Proyectos de redes y Sistemas de cómputo. CITMATEL, La Habana, Dic 2021. 53 PP.

## **SOBRE LOS AUTORES**

Guillermo Lastre Olazábal, graduado de Ingeniería en Máquinas Computadoras en el Instituto Superior Politécnico "Jose Antonio Echevarría" (ISPJAE), en 1990. Especialista en proyectos de redes, CCTV y servicios técnicos. Actualmente se desempeña como Director de la UEB de Proyectos de Redes y Sistemas de Cómputo de la Empresa de Tecnologías de la Información y Servicios Telemáticos (CITMATEL). Miembro de la Unión de Informáticos de Cuba (UIC). Identificador Orcid: <https://orcid.org/0000-0001-9716-0227>.

Gilbert Rafael García Cabrera, graduado de la carrera de Ciencias de la Computación en el año 2020 en la facultad de Matemática y Computación de la Universidad de La Habana. Vinculado a la Empresa de Tecnologías de la Información y Servicios Telemáticos Avanzados (CITMATEL) desde las prácticas laborales y posteriormente en el proyecto de Tesis de Grado. Trabaja desde el 2020 en el Grupo de Desarrollo e Innovación de la UEB Proyecto e Instalación de Redes y Sistemas de Cómputo de la empresa CITMATEL donde sus principales tareas son la instalación, implantación, mantenimiento y administración de aplicaciones informáticas para servicios de la empresa como el Servicio de Videokonferencia por Software Galy Meet, la Nube Computacional del CITMA, la Central Telefónica IP Issabel y la Cartelería Digital. Identificador Orcid: <https://orcid.org/0000-0002-2836-9147>

## **CONFLICTO DE INTERESES**

Este trabajo no presenta ningún conflicto de interés.

## CONTRIBUCIONES DE LOS AUTORES

Guillermo Lastré Olazábal: conceptualización: 100%, preparación: 90%, creación:100%, desarrollo del artículo: 100%, revisión crítica de cada una de las versiones del borrador del artículo: 100% y aprobación de la versión final a publicar: 100%, contribución a la idea: 100% y organización del artículo: 100%.

Gilbert Rafael García Cabrera: conceptualización: 80%, preparación: 70%, creación:100%, desarrollo del artículo: 90%, revisión crítica de cada una de las versiones del borrador del artículo: 100% y aprobación de la versión final a publicar: 100%, contribución a la idea: 100% y organización del artículo: 90%.