

# COMPARATIVA DE ENTORNOS OPEN-SOURCE PARA SISTEMAS DE SUPERVISIÓN APLICABLES A SMART GRIDS/SMART MICRO-GRIDS

José María Portalo Calero, Isaías González Pérez, Antonio José Calderón Godoy, Manuel Calderón Godoy  
Escuela de Ingenierías Industriales, Universidad de Extremadura, jportalo@alumnos.unes.es, {igonzp, ajcalde, calgodoy}@unex.es

## Resumen

*La supervisión de Smart Grids y Smart Micro-Grids es una función vital para conseguir una operación fiable y óptima. Por su parte, el movimiento open-source está recibiendo una creciente atención en la ciencia y la academia. Este artículo proporciona una comparativa de entornos software open-source para implementar sistemas de supervisión. Se exponen sus principales características (comunicaciones soportadas, sistemas operativos compatibles, etc.) y se discuten las funcionalidades más relevantes.*

**Palabras clave:** Smart Grid, Smart Micro-Grid, SCADA, open-source.

## 1 INTRODUCCIÓN

Las herramientas open-source son mejoradas continuamente además de incorporadas a marcos innovadores. Por ejemplo, la cantidad de dispositivos en el Internet de las Cosas (IoT, Internet-of-Things) puede incrementarse gracias a la tecnología open-source [4]. El despliegue de sensores y actuadores también puede ser facilitado por herramientas de tipo open-source [3, 18]. Para el paradigma de la Industria 4.0, los proyectos open-source son considerados como aceleradores clave en la adopción del IoT [17].

En el contexto de las energías renovables, los recursos open-source también están siendo utilizados como herramientas versátiles en diferentes aplicaciones. Por ejemplo, hardware open-source es empleado para medida y adquisición de magnitudes en [2] para la monitorización de una pila de combustible de hidrógeno.

En este sentido, la transformación digital de las redes energéticas da lugar al concepto de Smart Grids (SGs). Éstas son concebidas como redes inteligentes con capacidades mejoradas debido a la integración de sensores, automatización y supervisión [6, 20]. Desde el punto de vista energético, estas redes pueden combinar energías renovables e hidrógeno para lograr una operación medioambientalmente

sostenible y una optimización de la generación y distribución. Además, las Smart Micro-Grids (SMGs) son consideradas SGs a pequeña escala que pueden actuar de forma autónoma o conectada a la red de distribución [16], teniendo un enorme potencial para aumentar la penetración de las energías renovables [21].

Para SGs/SMGs, las herramientas open-source también son aplicadas como demuestran recientes publicaciones. En [1] y [5] tareas de medida y sensorización son realizadas por equipos open-source, mientras que en [6] una suite open-source es aplicada con éxito para implementar un sistema de monitorización.

Cabe enfatizar que algunas ventajas importantes de los sistemas open-source son configurabilidad abierta, facilidad de uso, bajo coste y conocimiento compartido (código, esquemas, etc.), las cuales motivan su amplio uso en diferentes dominios. A nivel de hardware, los denominados Single Board Computers (SBCs) constituyen ejemplos de dispositivos open-source, siendo Arduino y Raspberry Pi casos ilustrativos. El primero de ellos es una plataforma de desarrollo electrónico basada en un microcontrolador, mientras que el segundo corresponde a un microprocesador de mayores capacidades y sistema operativo embebido.

Con respecto al software open-source, el desarrollador posee mayor control sobre el sistema y los datos, y configurabilidad de más bajo nivel [19]. Entre la gran cantidad de paquetes disponibles, el presente trabajo pone el foco en suites dedicadas a diseñar e implementar sistemas de supervisión, generalmente conocidos por el término en inglés Supervisory Control and Data Acquisition (SCADA). La Figura 1 muestra el diagrama de bloques general de un sistema de supervisión aplicado a un proceso que es sensorizado y automatizado. En este esquema, un dispositivo de adquisición de datos es empleado para recoger datos del proceso mediante sensores e, incluso, aplicar comandos de control a través de actuadores. Este dispositivo se conecta al sistema SCADA utilizando una red de comunicaciones a fin de establecer un intercambio continuo de datos. La

información compartida es mostrada al usuario/operador para llevar a cabo la vigilancia del comportamiento del proceso. Además, dicha información es almacenada para su tratamiento y análisis.



Figura 1: Diagrama de bloques general de sistema de supervisión aplicado a proceso

El objetivo de este artículo es mostrar los entornos open-source que pueden aplicarse para supervisar infraestructuras avanzadas como SGs/SMGs. Así mismo, se realiza una comparativa de sus características principales. Esta información resulta útil para científicos, académicos y profesionales que necesiten elegir la opción más ventajosa de entre las disponibles.

Este trabajo se enmarca en un proyecto de investigación que versa sobre el desarrollo de una réplica digital de una SMG que integra energía fotovoltaica, generadores de hidrógeno y pilas de combustible. En este escenario, el manejo adecuado de señales de sensores y actuadores de la SMG mediante un sistema de supervisión adquiere una importancia vital para el seguimiento en tiempo real de la operación de la red así como para el posterior procesamiento de los datos. En el mencionado proyecto, herramientas tipo open-source están incorporándose debido a los beneficios que aportan.

La estructura del resto del documento es como sigue. La sección 2 expone una serie de suites open-source orientadas a tareas de supervisión. La tercera sección lleva a cabo una comparación de sus principales características. Por último, las conclusiones del trabajo son comentadas.

## 2 SISTEMAS SCADA OPEN-SOURCE

Un sistema SCADA principalmente realiza tres acciones, adquisición de datos, monitorización y control. Los datos del dispositivo de campo son enviados a la unidad de control central principal, y esta administra y controla los datos y permite la supervisión del sistema. Estos sistemas se utilizan principalmente en la industria para procesos industriales. Un sistema SCADA puede integrarse con controladores como PLC (Programmable Logic

Controller) y DCS (Distributed Control System) ubicados en lugares remotos. Además, los sistemas SCADA también se aplican a SG y SMG, siendo una parte fundamental de su operación.

Para elegir un software SCADA debe valorarse la compatibilidad con los diferentes sistemas operativos, qué controladores admite, cuál es su coste, si es escalable, actualizable, seguro y si va a tener soporte posterior.

Este trabajo se centra en la opción de los entornos software SCADA de naturaleza open-source, y se valoran según sus características más relevantes. Entre las diferentes versiones que nos podemos encontrar como opción adecuada a nuestras necesidades, hay que valorar según la plataforma en la que vayamos a instalar dicho software. Aunque en general casi todos son multiplataforma o basados en web, nos encontraremos con esa limitación en algunos entornos.

Hay software SCADA para PC, Macintosh y SBCs. En cuanto al sistema operativo también vamos a poder elegir. Aunque casi todas las aplicaciones software son para diferentes distribuciones Linux (Ubuntu, Debian, CentOS, Raspbian, etc.), vamos a tener versiones en las diferentes opciones de Windows, MacOS, Android OS, también entornos basados en web, accesibles incluso desde internet.

Otro aspecto importante a tener en cuenta es la gama de protocolos de comunicación de que dispone cada paquete. Algunos de los protocolos compatibles son Modbus RTU (Remote Terminal Unit), Modbus TCP/IP (Transmission Control Protocol/Internet Protocol), BACnet (Building Automation and Control Networks), OneWire, DNP3 (Distributed Network Protocol), IEC 101, OPC DA (Open Platform Communications – Data Access), OPC UA (Open Platform Communications – Unified Architecture), MQTT (Message Queue Telemetry Transport), RFC (Remote Function Call), ASCII (American Standard Code for Information Interchange) Serial, HTTP (Hypertext Transfer Protocol), conectores SQL (Search Query Language) para MySQL, SQLite, ODBC (Open Database Connectivity), etc.

El entorno de desarrollo o lenguaje de programación de las aplicaciones también es muy diverso. Al ser software open-source, los desarrolladores han optado por lenguajes de uso común. Principalmente son Java, C, C++, Python, PHP, Free Pascal, Javascript, etc.

En los diferentes sitios web de las aplicaciones SCADA, podemos ver sus características más interesantes. A continuación se comentan las

características relevantes de siete suites SCADA tipo open-source.

### 2.1 IndigoSCADA

IndigoSCADA [11] es un sistema SCADA de tamaño reducido desarrollado completamente en C y C++ con soporte para múltiples sistemas operativos y múltiples controladores de protocolo *front-end*. La Figura 2 muestra el aspecto de la interfaz de este entorno.

Casi todas las funciones del software SCADA son de código abierto y aceptan controladores de protocolo maestro-esclavo. El interfaz hombre-máquina (Human-Machine Interface, HMI) admite múltiples ventanas e incorpora tipos de datos IEC 60870-5-101. Los protocolos admitidos son OPC DA, DNP 3.0, RFC 1006, Modbus, MQTT 3.1.1. Por su parte, los sistemas operativos soportados son Windows y Linux. La licencia de uso es GPL (General Public License).

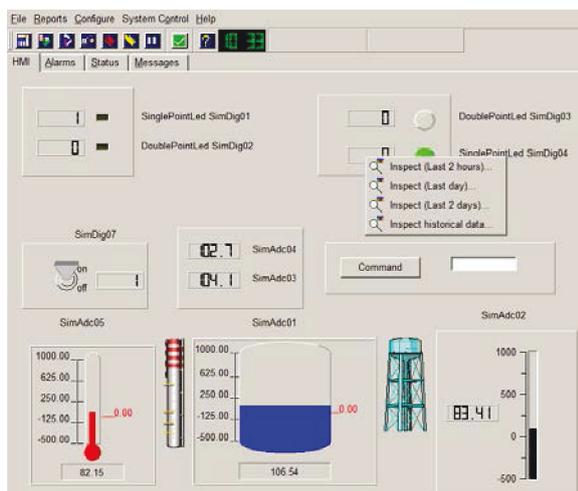


Figura 2: Ejemplo de interfaz de IndigoSCADA

### 2.2 OpenSCADA

OpenSCADA [8] es una plataforma independiente basada en Java, con un diseño de sistema moderno que proporciona seguridad y flexibilidad al mismo tiempo. La comunicación se basa en cliente/servidor mediante TCP/IP.

La Figura 3 muestra el aspecto de la interfaz de este software. Cabe indicar que OpenSCADA no es una solución lista para usar, sino un conjunto de herramientas que se pueden combinar de muchas maneras diferentes. Proporciona bibliotecas de desarrollo, aplicaciones de interfaz, herramientas de configuración masiva, aplicaciones *front-end* y *back-end*. Respecto a los sistemas operativos soportados, éstos son Windows y Linux.

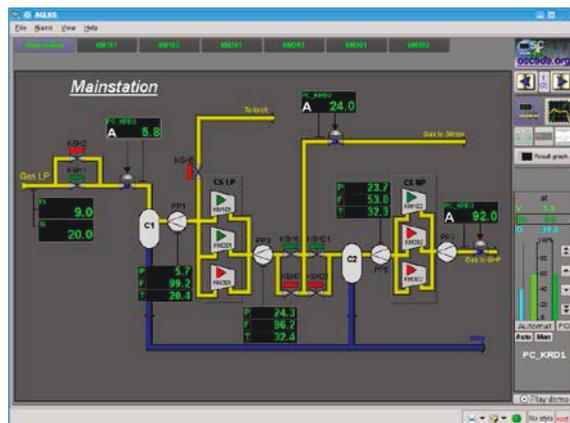


Figura 3: Aspecto del interfaz de OpenSCADA

### 2.3 Eclipse NeoSCADA

Eclipse NeoSCADA [13] (anteriormente llamado Eclipse SCADA) es un proyecto de Eclipse Incubator creado en julio de 2013, que tiene como objetivo proporcionar un sistema SCADA de código abierto completo y de vanguardia que se pueda utilizar de forma inmediata o como una plataforma para crear. Procede del proyecto OpenSCADA, que ahora proporciona una funcionalidad adicional a Eclipse NeoSCADA.

Este entorno admite los protocolos JDBC (Java Database Connectivity), Modbus TCP/IP y RTU, RFC 1006, etc. Los módulos de protocolos OPC y SNMP (Simple Network Management Protocol) están disponibles fuera de Eclipse SCADA, es decir, son de pago. Eclipse SCADA es un sistema de código abierto, bajo licencia EPL (Eclipse Public License).

### 2.4 RapidSCADA

RapidSCADA [10] es un software gratuito con el que se pueden crear sistemas SCADA para de automatización industrial, sistemas domóticos, sistemas de contabilidad energética, sistemas de seguridad y alarma contra incendios, sistemas de control de acceso, etc.

Dispone de funcionalidad completa de forma gratuita y no hay límites de tiempo. Cuenta con acceso vía web y un interfaz intuitivo.

### 2.5 ScadaBR

ScadaBR [12] es un sistema SCADA que ofrece funciones de registro, informes y control. Permite definir las vistas gráficas de la planta monitorizada. En las vistas, el usuario puede definir botones, imágenes animadas, imágenes, alarmas. La Figura 4 ilustra la interfaz de esta suite.

Los protocolos utilizables para adquisición de datos son: Modbus RTU, Modbus TCP/IP, ASCII serial, BACnet, OneWire, HTTP, Zwave, OPC, DNP3, etc. Por su parte, los sistemas operativos soportados son Windows, Linux y MacOS, La licencia es GPL. Necesita instalar previamente Java SE Runtime Environment 6u45. Permite trabajar con bases de datos MySQL o Derby.

Entre sus características más reseñables están el ambiente de desarrollo para aplicaciones SCADA, acceso desde navegador (escritorio, móvil), API (Application Programming Interface) con ejemplos en Java, Visual Basic, PHP y Python, entre otros, etc. Existe una demo del programa accesible desde [14].

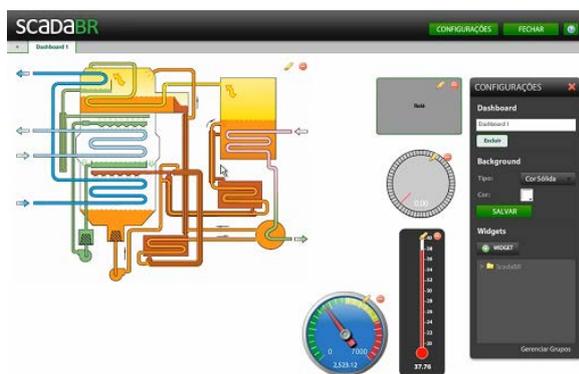


Figura 4: Ilustración del interfaz de ScadaBR

## 2.6 SCADA-LTS

SCADA-LTS [9] es una solución multiplataforma de código abierto, basada en web para crear cualquier sistema SCADA. Su arquitectura de software le define porque está desarrollado en Java: el servidor se ejecutará en cualquier arquitectura (Windows/MacOS/Linux), distribuido en WAR (*Web Application Resource*) (multiplataforma) y Windows Installer, y la interfaz de usuario se ejecuta desde un navegador web estándar. No se necesita instalación del cliente.

Utiliza APIs de SOAP (*Simple Object Access Protocol*) y REST (*Representational State Transfer*) para integración personalizada, motor de adquisición de datos (para muchos protocolos populares de inserción y sondeo), generador de vistas gráficas (HMI), informes de datos con gráficos, multilingüe con inglés, portugués, español, etc.

En cuanto a protocolos de comunicación cabe destacar Modbus TCP/IP, DNP3, IEC 101, OPC DA 2.0, Lectores ASCII Serial, HTTP, conectores SQL (extraer datos de bases de datos SCADA heredadas y otros sistemas).

## 2.7 Tango Controls

Tango Controls [15] es un conjunto gratuito de herramientas de control de código abierto orientado a dispositivos para controlar cualquier tipo de hardware o software y construir sistemas SCADA. Tango Controls es compatible con C++, Java y Python para todos los componentes.

El conjunto de herramientas se desarrolló principalmente para las necesidades de los centros de investigación, pero la idea y el concepto (filosofía) subyacentes era crear un entorno (*framework*). Tango Controls se puede usar no solo como un kit de herramientas de controles orientado a dispositivos y escribir sus propias aplicaciones, sino que también puede usarse como producto final. Tango Controls es un conjunto de herramientas de hardware independiente, lo cual implica que se puede usar cualquier controlador para conectar hardware con Tango Controls.

Los controles de Tango se pueden usar como un sistema de escritorio, web (remoto o local) o distribuido. Es orientado a objetos y se puede usar para sistemas DCS, aplicaciones del IoT, comunicaciones máquina a máquina (*Machine-to-Machine*, M2M), etc.

## 3 DISCUSIÓN

Después de exponer las características principales de los diferentes paquetes de software open-source para supervisión, vamos a valorarlos según diferentes aspectos a tener en cuenta en el momento de elegir uno u otro software, como son la disponibilidad de nuevas versiones y actualizaciones, facilidad y opciones de instalación, complementos adicionales, posibilidad de uso en entornos virtuales, aplicaciones de demostración, versatilidad, etc. La Figura 5 ilustra la comparativa de las características más relevantes de los entornos open-source considerados en la anterior sección.

Una de las características más valoradas puede ser la compatibilidad con dispositivos de diferentes proveedores, y sobre todo compatibles con nuestras instalaciones, que vienen demostradas por la propia variedad y conectividad mediante los diferentes protocolos de comunicación más utilizados en la industria. En cuanto a esto, casi todos son compatibles con los principales modos de intercambiar información, esto es, los protocolos más extendidos como OPC, Modbus, etc.

Aunque estas aplicaciones se encuadran dentro del movimiento open-source, esto no implica que sean gratuitos, por lo que en algunos paquetes hemos

encontrado que si queremos características adicionales o más especiales, hay que acudir a fórmulas de pago. Este es el caso de RapidSCADA, que ciertos complementos tienen un coste adicional, como por ejemplo sus *plugins* “Chart Pro”, “Dashboard”, “Elastic Report”, entre otros.

Comparativa SCADA	IndigoSCADA	OpenSCADA	Eclipse SCADA	RapidScada	SCADABR	SCADA-LTS	Tango-Controls
Sistema Operativo							
Arquitectura Cliente/Servidor	Si	Si	Si	Si	Si	Si	Si
Entorno WEB		Si	Si	Si	Si	Si	Si
Lenguaje Programación	C, C++	Java		.NET	Java	Java	Java, C++, PHP
Conjunto de Herramientas		Si	Si				Si
Licencia	GPL		EPL		GPL		
OPC	Si	Si		Si	Si	Si	
Modbus (TCP, RTU)	Si	Si	Si	Si	Si	Si	
MQTT	Si			Si			
HTTP		Si		Si	Si	Si	Si

Figura 5: Comparativa de características de los entornos open-source considerados

También hay que tener en cuenta la posibilidad de acceso web, que nos permitiría utilizarlo desde cualquier parte, así como la existencia de comunidades de soporte, foros, wikis, etc. Donde podamos acudir siempre que tengamos tanto dudas como si queremos hacer aportaciones de nuestras experiencias, incluso dar soporte nosotros mismos. En eso consiste la comunidad open-source.

Siempre que se realiza una inversión en equipamiento, tanto hardware como software, se valora que sea novedoso e innovador, que no sea un producto caducado, pero también que tenga actualizaciones regulares y recientes. En este aspecto hemos podido comprobar que algunos paquetes de software, están actualizados hasta la fecha actual, Mayo 2019, como es el caso de OpenSCADA, Rapid SCADA, IndigoSCADA o Tango-Controls y en otros casos sus últimas actualizaciones son más antiguas, como SCADA-LTS. Esto no quiere decir obligatoriamente que los programas estén descuidados en sus actualizaciones, sino que puede ser que no tengan fallos y por eso no es necesario actualizaciones parciales, y se centren en actualizaciones más globales y que afecten en mayor medida a todo el entorno. Así encontraremos paquetes que hacen uso del término LTS (Long Term Support), como OpenSCADA y SCADA-LTS, que nos indica que sus actualizaciones globales son más lentas y que han sido diseñadas para tener soportes durante un período más largo que el normal.

Hay que mencionar que aunque la mayoría de las aplicaciones están diseñadas para su instalación en entornos Linux, que se presta más a la filosofía open-

source, también hay versiones tanto para Windows, como para MacOS. Pero cabe destacar que algunos tienen versiones para placas microcontroladoras como Raspberry Pi u Orange Pi, como son OpenSCADA, RapidSCADA y Tango-Controls. Incluso OpenSCADA tiene una versión para Android.

Todos tienen una página web de exposición del producto, donde nos muestran las características más importantes de los mismos. Pero, además, algunos poseen repositorios, donde ubican todas las diferentes versiones, como es el caso de ‘Github’, que es una plataforma de desarrollo, donde pueden almacenar y revisar código, manejar proyectos, y elaborar software utilizado por millones de desarrolladores. Hemos encontrado algunos que, además de la ayuda de la comunidad, disponen de elaborados manuales, tanto en apartados de una página web, el caso de OpenSCADA, como en documentos pdf, en el caso de IndigoSCADA.

Un aspecto destacable es que nos hemos encontrado con el caso de ScadaBR, que se inició en 2006 como un proyecto independiente, y ha ido convergiendo, hasta llegar a una versión, ScadaBR 2.X, que desciende de otra, o más bien es una versión adaptada de esta otra, que es SCADA-LTS.

Todos los entornos considerados tienen una instalación con acceso web, a excepción de IndigoSCADA, que es una instalación local en un ordenador, aunque se base, como los demás, en la arquitectura cliente/servidor. El tener acceso web nos permite su uso desde diferentes ubicaciones, sin necesidad de instalar software adicional, simplemente con el navegador web adecuado. Así, algunos disponen de distribuciones WAR, como SCADA-LTS, que nos permite usar el sistema simplemente con cargar estos ficheros bajo el directorio correspondiente de un servidor web. En otros casos, en cambio, hay que hacer la instalación más elaborada, aunque suele ser muy fácil, por los *scripts* de instalación de que disponen, como el caso de OpenSCADA.

Una opción muy interesante consiste en la posibilidad de disponer de la aplicación demo en un entorno web, como OpenSCADA, y RapidSCADA, una versión *live* para ejecutarla desde un USB o un DVD, como OpenSCADA, así como disponer de una máquina virtual completamente instalada y configurada para usarla bajo VirtualBox. Este es el caso de Tango Controls. Otra posibilidad parecida a esta última, nos la ofrece SCADA-LTS, que tiene una distribución para instalarla sobre Docker, que es un contenedor (*container*) más liviano y rápido que una virtualización.

Por último, cabe reseñar que, además de los ya considerados, existen otros paquetes para supervisión open-source, como son OpenAPC, Seer2, Szarp, Mango Automation, AdvancedHMI, Visual Tcl, o Pvbrowser.

## 4 CONCLUSIONES

La supervisión de SGs y SMGs resulta crítica para su correcto funcionamiento. Así mismo, el movimiento open-source está ganando cada vez una mayor presencia en la literatura científica. Dado tal contexto, este trabajo ha presentado una comparativa de diversos paquetes software open-source para implementar sistemas SCADA. Se han expuesto sus principales características en cuanto a conectividad, sistemas operativos soportados, etc.

Se espera que este documento resulte útil para científicos, académicos y profesionales en el ámbito de la supervisión que requieran seleccionar la mejor opción de entre los entornos software disponibles.

Trabajos futuros se centran en evaluar la operación a largo plazo de estos sistemas open-source teniendo en cuenta los requisitos de fiabilidad impuestos por las SGs/SMGs.

### Agradecimientos

Este trabajo ha sido financiado por el proyecto IB18041 de la Junta de Extremadura en el VI Plan Regional de I+D+i (2017-2020), cofinanciado por el Fondo Europeo de Desarrollo Regional FEDER (Programa Operativo FEDER de Extremadura 2014–2020).

### English summary

## COMPARISON OF OPEN-SOURCE ENVIRONMENTS FOR SUPERVISORY SYSTEMS APPLIED TO SMART GRIDS/SMART MICRO-GRIDS

### Abstract

*Supervision of Smart Grids and Smart Micro-Grids is a vital function to achieve a reliable and optimal operation. On the other hand, open-source movement is receiving an increasing attention in science and academy. This paper provides a comparison of open-source software environments to implement supervisory systems. Their main features are*

*expounded (supported communications, compatible operating systems, etc.) and the most relevant functionalities are discussed.*

**Keywords:** Smart Grid, Smart Micro-Grid, SCADA, open-source.

### Referencias

- [1] Batista, N.C., Melício, R., Mendes, V.M.F., (2014) “Layered Smart Grid architecture approach and field tests by ZigBee technology”, *Energy Conversion and Management*, vol. 88, pp. 49-59.
- [2] Calderón, A.J., González, I., Calderón, M., Segura, F., Andújar, J.M., (2016) “A New, Scalable and Low Cost Multi-Channel Monitoring System for Polymer Electrolyte Fuel Cells”, *Sensors*, vol. 16(3), 349.
- [3] Costa, D.G., Duran-Faundez, C., (2018) “Open-Source Electronics Platforms as Enabling Technologies for Smart Cities: Recent Developments and Perspectives”, *Electronics*, vol. 7, pp. 404.
- [4] Fisher, R., Ledwaba, L., Hancke, G., Kruger, C., (2015) “Open Hardware: A Role to Play in Wireless Sensor Networks?”, *Sensors*, vol. 15, pp. 6818-6844.
- [5] Fuentes, M., Vivar, M., Burgos, J.M., Aguilera, J., Vacas, J.A., (2014) “Design of an accurate, low-cost autonomous data logger for PV system monitoring using Arduino™ that complies with IEC standards”, *Solar Energy Materials & Solar Cells*, vol. 130, pp. 529-543.
- [6] González, I., Calderón, A.J., Andújar, J.M., (2017) “Novel Remote Monitoring Platform for RES-Hydrogen based Smart Microgrid”, *Energy Conversion and Management*, vol. 148, pp. 489-505.
- [7] González, I., Calderón, A.J., Figueiredo, J., Sousa, J.M.C., (2019) “A Literature Survey on Open Platform Communications (OPC) Applied to Advanced Industrial Environments”, *Electronics*, vol. 8, pp. 510.
- [8] <http://oscada.org/main/download/>
- [9] <http://scada-lts.org/>
- [10] <https://rapidscada.org/>

- [11] <https://sourceforge.net/projects/indigoscada/files/latest/download>
- [12] <https://sourceforge.net/projects/scadabr/>
- [13] <https://www.eclipse.org/eclipsescada/>
- [14] <https://www.ege3d.it/scada2d.php>
- [15] <https://www.tango-controls.org/what-tango-controls/>
- [16] Koochi-Kamali, S., Rahim, N.A., (2016) “Coordinated control of smart microgrid during and after islanding operation to prevent under frequency load shedding using energy storage system”, *Energy Conversion and Management*, vol. 127(1), pp. 623-646.
- [17] Martínez, B., Vilajosana, X., Kim, I.H., Zhou, J., Tuset-Peiró, P., Xhafa, A., Poissonnier, D., Lu, X., (2017) “I3Mote: An Open Development Platform for the Intelligent Industrial Internet”, *Sensors*, vol. 17, pp. 986.
- [18] Mejías, A., Reyes, M., Márquez, M.A., Calderón, A.J., González, I., Andújar, J.M., (2017) “Easy handling of sensors and actuators over TCP/IP Networks by Open Source Hardware/Software. *Sensors*, vol. 17, 94.
- [19] Wong, B.P., Kerkez, B., (2016) “Real-time environmental sensor data: An application to water quality using web services”, *Environmental Modelling & Software*, vol. 84, pp. 505-517.
- [20] Yang, Q., An, D., Yu, W., Tan, Z., Yang, X., (2016) “Towards Stochastic Optimization-Based Electric Vehicle Penetration in a Novel Archipelago Microgrid”, *Sensors*, vol. 16, pp. 907.
- [21] Zhang, L., Gari, N., Hmurcik, L.V., (2014) “Energy management in a microgrid with distributed energy resources”, *Energy Conversion and Management*, vol. 78, pp. 297-305.



© 2019 by the authors.  
Submitted for possible  
open access  
publication under the  
terms and conditions of the Creative Commons  
Attribution CC BY-NC-SA 4.0 license  
(<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).