

Laboratorio Remoto para la Formación de Usuarios Basado en el Cloud

Pablo Godoy¹, Ricardo Cayssials², Carlos García Garino³

¹ITIC, FCEN and Facultad de Ingeniería, Universidad Nacional de Cuyo, Mendoza, Argentina

²Universidad Tecnológica Nacional - FRBB, Bahía Blanca, Buenos Aires, Argentina

³ITIC and Facultad de Ingeniería, Universidad Nacional de Cuyo, Mendoza, Argentina

{pgodoy, cgarcia}@itu.uncu.edu.ar, rcayssials@frbb.utn.edu.ar

Resumen

Este artículo presenta un prototipo de laboratorio remoto de redes de sensores inalámbricos destinado a la formación y entrenamiento de usuarios. El prototipo ha sido desplegado sobre una infraestructura de Cloud Computing, con el objetivo de aprovechar los beneficios que esta tecnología puede brindar a los laboratorios remotos. Se analiza el estado del arte del problema, el cual permite especificar las características necesarias que debe cumplir un laboratorio remoto destinado a formación y entrenamiento de usuarios. Luego se discute la implementación de los módulos principales que componen el laboratorio remoto, y el análisis que conduce a escoger un despliegue basado en Cloud Computing.

El trabajo presenta resultados de experimentos diseñados para validar la tecnología propuesta.

Palabras clave: Laboratorios remotos, entrenamiento de usuarios, WSN, Cloud Computing.

Abstract

This paper presents a wireless sensor networks (WSN) remote laboratory prototype aimed at user training. The prototype has been deployed on a Cloud Computing infrastructure, in order to take advantage of the benefits that this technology can provide to remote laboratories. The state of the art of the problem is analyzed. This analysis allows to specify the features required for a remote laboratory intended to be applied in user training. Then, the implementation of the main modules that compose the remote laboratory, and the analysis that led to chose a deployment based on Cloud Computing are discussed.

This paper also presents experiments aimed to validate the proposed technology.

Keywords: Remote laboratory, user training, WSN, Cloud Computing.

1. Introducción

Los laboratorios remotos son plataformas que permiten el acceso remoto a diferentes tipos de equipos, dispositivos, laboratorios, etc. a través de Internet. Son sistemas complejos que incluyen un gran número de componentes. Pueden estar destinados a investigación científica, desarrollo de aplicaciones, formación de usuarios, enseñanza, entre otras aplicaciones.

Están compuestos principalmente por dos sistemas:

- El sistema bajo prueba (equipamiento, máquinas eléctricas, dispositivos electrónicos, un laboratorio físico, etc.).
- El sistema de acceso remoto, que permite a los usuarios acceder al sistema bajo prueba para realizar experimentos remotamente.

El sistema de acceso remoto está formado por diferentes módulos de hardware y software, entre estos:

- Sistema de interconexión (entre el sistema bajo prueba y el resto del laboratorio remoto).
- Autenticación de usuarios.
- Sistema de reserva de turnos.
- Base de datos.
- Interface web.

Los laboratorios remotos tienen el potencial de llegar a ser una importante herramienta para la enseñanza a través de los cursos conocidos en la bibliografía como MOOCs (cursos masivos abiertos en línea) [1, 2].

Las redes de sensores inalámbricos (WSNs) están formadas por dispositivos denominados nodos que tiene la capacidad de detectar y sensar variables, procesar y almacenar datos y comunicarse inalámbricamente. Una WSN puede incluir decenas o cientos de nodos [3].

Existen un gran número de aplicaciones para las WSN, incluyendo: monitoreo del medio ambiente, cuidado de la salud, automatización de edificios, defensa, agricultura, etc. [4, 5].

Cloud Computing es un modelo que permite acceso ubicuo y bajo demanda a un conjunto de recursos computacionales configurables, virtualizados y compartidos (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios), que pueden ser proporcionados y liberados rápidamente y con mínimo esfuerzo de gestión y casi ninguna interacción con el proveedor de servicios de Cloud Computing. Estos recursos son ofrecidos por proveedores de servicios de Cloud Computing a clientes externos a través de Internet como servicios web, basado en acuerdos negociados entre el proveedor de servicios y los clientes [6, 7].

Un problema no resuelto citado en la bibliografía es proporcionar a los laboratorios remotos características tales como calidad de servicio, seguridad y almacenamiento de datos confiable [8]. Estos requisitos se vuelven más importantes en laboratorios remotos donde los estudiantes necesitan ser monitoreados y evaluados por los profesores. Actualmente, los proveedores de servicios de Cloud Computing ofrecen características tales como altos niveles de calidad de servicio, seguridad y almacenamiento de datos confiable (copias de seguridad, protección contra intrusos, etc.). Por lo tanto, la implementación de algunos componentes de los laboratorios remotos a través de servicios de Cloud Computing puede proporcionar estas características a los laboratorios remotos.

En este trabajo se describe la construcción de un prototipo de laboratorio remoto de WSN basado en el Cloud destinado a actividades de formación y experimentación académica. El resto del trabajo se organiza de la siguiente manera. La Sección 2 resume y analiza los trabajos relacionados. La sección 3 describe las características que requiere un laboratorio remoto para ser aplicado en la formación de usuarios. La sección 4 presenta un modelo de implementación de laboratorios remotos en una infraestructura de Cloud Computing, y la sección 5 describe como se han adaptado diferentes módulos del laboratorio remoto para su aplicación en tareas de formación de usuarios. La sección 6 describe experimentos realizados para validar el modelo propuesto, y finalmente la sección 7 presenta las conclusiones y trabajos futuros.

2. Trabajos Relacionados

En los últimos años se han desarrollado un gran número de laboratorios remotos de WSN. Éstos fueron implementados en sus inicios con la finalidad de permitir a investigadores y desarrolladores implementar componentes de software y protocolos de comunicaciones para WSN. Los trabajos de Steyn y Hancke [9] y El-Darymli *et al.* [10] presentan los estudios del estado del arte sobre laboratorios remotos de WSN más completos

hallados en la literatura. Otros trabajos que incluyen estudios del estado del arte sobre laboratorios remotos de WSN son [11, 12].

Actualmente la integración de laboratorios remotos a Cloud Computing ha sido estudiada por pocos autores [13, 14].

Hay algunos trabajos sobre la integración de WSN a entornos de Cloud Computing [15, 16, 17], los cuales presentan mecanismos para compartir los datos recolectados por las WSN a través de tecnologías de Cloud Computing. De esta manera se aprovecha la elevada capacidad de almacenamiento de datos del Cloud y la alta disponibilidad y robustez que presenta.

Algunos autores han propuesto plataformas basadas en el Cloud destinadas a tareas de enseñanza sobre temas relacionados con redes de computadoras o ciberseguridad [18, 19, 20]. Los equipos bajo prueba consisten en máquinas virtuales provistas por un proveedor de servicios de Cloud Computing sobre la cual los estudiantes pueden realizar sus experimentos. Estos trabajos muestran la utilidad del Cloud para tareas de enseñanza en temas relacionados con computación. Puede observarse la necesidad de que el proveedor de servicios de Cloud Computing brinde servicios del tipo IaaS, ya que el alumno realiza los experimentos sobre las máquinas virtuales o recursos computacionales virtualizados provistos por el proveedor de servicios Cloud.

A diferencia de los trabajos mencionados anteriormente, este documento propone un modelo para la implementación de un laboratorio remoto de WSN usando servicios de Cloud Computing, en el cual el equipo bajo prueba (una WSN) no es un recurso virtualizado por en el Cloud, sino que es externo al mismo. El laboratorio remoto propuesto puede implementarse en el Cloud usando servicios del tipo IaaS o PaaS.

3. Laboratorios Remotos Para Actividades de Formación

3.1. Tipos de Laboratorios Remotos según su Aplicación

Un estudio del estado del arte permite diferenciar dos tipos diferentes de laboratorios remotos según su aplicación [21]:

- Laboratorios remotos destinados a desarrollo e investigación.
- Laboratorios remotos destinados a formación y entrenamiento de usuarios.

La diferencia más importante entre los mismos es el nivel de conocimientos requerido por parte del usuario acerca del equipo bajo prueba. Los usuarios de los laboratorios remotos destinados a investigación necesitan un alto nivel de conocimientos del equipo bajo prueba para poder

utilizarlos. Estos usuarios pueden realizar experimentos sin ayuda de ninguna otra persona. Los laboratorios remotos orientados a investigación permiten a los usuarios acceder a los recursos del laboratorio sin ninguna restricción. Por ejemplo, un laboratorio remoto de WSN orientado a investigación, como MoteLab [22], permite a los usuarios cargar programas o sistemas operativos en los nodos de la WSN sin ningún tipo de restricción.

Los laboratorios remotos destinados a propósitos de formación de usuarios o educativos requieren de los usuarios niveles de conocimientos variados, pudiendo ser bajos o intermedios. Por ejemplo, Marianetti [23] propone un laboratorio remoto académico para la enseñanza de dispositivos lógicos programables (PLD). En este laboratorio remoto, usuarios con bajos niveles de conocimientos son guiados a través experimentos predefinidos y restringidos. Por lo general, los estudiantes no pueden ejecutar cualquier programa sobre el laboratorio remoto, sólo se les permite añadir bloques de código o modificar variables. Debido al bajo nivel de conocimiento de los usuarios, los laboratorios remotos destinados a formación de usuarios requieren sistemas de protección más robustos para evitar posibles daños provocados por configuraciones erróneas.

Además, en los laboratorios remotos destinados a tareas de enseñanza puede ser necesario realizar seguimiento de las actividades realizadas por los estudiantes, para verificar la realización de trabajos prácticos y exámenes. Como resultado, se requieren componentes adicionales para permitir el seguimiento de las actividades realizada por los usuarios [24]. Otra diferencia es la exactitud necesaria de los resultados. En el caso de un laboratorio remoto orientado a la formación de usuarios, puede no ser necesaria una alta precisión en los resultados. En cambio, un laboratorio remoto orientado a la investigación científica requiere la mayor precisión posible en los resultados.

Un laboratorio remoto orientado a investigación generalmente no requiere que parte del experimento esté predefinido. Por ejemplo, los usuarios de un laboratorio remoto de redes de computadoras orientado a investigación, usualmente desean cargar sus propios programas, sistemas operativos e incluso controladores de hardware. En cambio, un laboratorio remoto orientado a formación y entrenamiento requiere que parte de los experimentos esté predefinida, de modo que los usuarios no tengan que configurar el experimento desde cero, facilitándole así la realización de los mismos.

Tabla 1: Diferencias entre laboratorios remotos orientados a formación de usuarios y orientados a investigación y desarrollo

	Orientados a investigación y desarrollo	Orientados a formación de usuarios
Usuarios	Investigadores y desarrolladores	Usuarios que necesitan ser entrenados en el uso de una tecnología
Nivel de conocimientos requerido	Alto	Intermedio o bajo
Restricciones de acceso	Pocas o ninguna, los usuarios pueden realizar cualquier experimento	Si. Los usuarios no pueden realizar cualquier configuración
Experimentos predefinidos	No. Los usuarios necesitan cargar sus experimentos y configuraciones, usualmente desde cero	Los usuarios interactúan con experimentos predefinidos
Robustez de los sistemas de protección	Baja. Los usuarios poseen conocimientos elevados	Alta. Los usuarios con bajos conocimientos pueden cometer errores
Exactitud requerida de los resultados	Muy alta. Propósitos científicos	No muy alta, propósitos didácticos
Interfaces web	Diseño poco elaborado o no existe interfaz web	El diseño de la interface web debe permitir un uso simple, intuitivo y didáctico
Datos almacenados en las bases de datos	Resultados de los experimentos	Resultados de los experimentos y en algunos casos, la actividad realizada por los usuarios para permitir el seguimiento
Ejecución de los experimentos	Por lotes	Interactivos

Otra diferencia puede encontrarse en cuanto a la exactitud necesaria de los resultados. Si bien en el caso de un laboratorio remoto orientado a formación y entrenamiento los resultados deben ser cercanos a la realidad, puede no ser necesaria una gran exactitud, ya que el fin de los mismos es didáctico. En cambio, un laboratorio remoto orientado a investigación científica requiere la mayor

exactitud posible en cuanto a los resultados. La Tabla 1 resume las diferencias más importantes.

3.2.Laboratorios Remotos de WSN Actuales

Estudios del estado del arte publicados en la bibliografía sobre laboratorios remotos de WSN permiten conocer los componentes, módulos y características comunes de estos laboratorios remotos [9, 10, 11, 12]. Además, estos trabajos, junto con los analizados en la sección 3.1 permiten inferir qué componentes, módulos o características permiten o dificultan el uso de un laboratorio remoto de WSN en tareas de formación o entrenamiento de usuarios. A continuación se presenta una lista de los módulos que necesitan un diseño adecuado con el fin de utilizar el laboratorio remoto para tareas de formación usuarios:

- Operación en modo por lotes o en tiempo real: Es preferible una interacción en tiempo real para permitir una mayor interactividad con el equipo bajo prueba.
- Datos que se almacenan: Se debe almacenar información sobre la actividad realizada por los usuarios, con el fin de permitir realizar el seguimiento de las actividades de los usuarios, especialmente si el laboratorio remoto se utiliza en tareas de enseñanza.
- Interfaz web: Debe permitir un uso sencillo, intuitivo y didáctico del laboratorio remoto.
- Herramientas de análisis: Pueden ser de utilidad para ayudar a los usuarios a entender el funcionamiento de la WSN.

4. Implementación de un Laboratorio Remoto de WSN Basado en el Cloud

El objetivo principal del trabajo realizado es la implementación de un laboratorio remoto destinado a formación de usuarios. El primer prototipo implementado no se basó en el Cloud, sino que se utilizó un modelo tradicional para laboratorios remotos. Posteriormente, la lectura de trabajos recientes [8, 13, 14] motivó el análisis de las ventajas y desventajas de la integración del laboratorio remoto con tecnologías de Cloud Computing. La implementación de ambos prototipos permitió conocer características de cada uno que se mencionan a continuación.

4.1.Implementación Tradicional

Un laboratorio remoto tradicional está constituido por un ordenador que controla el equipo bajo prueba y permite el acceso remoto a través de una red LAN o Internet. La primera implementación del laboratorio remoto

presentado en este artículo se basó en el modelo de capas presentado por Marianetti [23]. El modelo en capas de Marianetti se presenta en la figura 1. La implementación del laboratorio remoto tradicional se describe en detalle en [25].

Desde el punto de vista de la implementación, la diferencia más importante entre un laboratorio remoto tradicional y uno basado en el Cloud se da en el sistema de interconexión. Este sistema permite interconectar el equipo bajo prueba (en este caso una WSN) con el resto del laboratorio remoto. Para el primer prototipo de laboratorio remoto de WSN, basado en un esquema tradicional, los componentes de hardware del sistema de interconexión entre el laboratorio remoto y el equipo bajo prueba consisten en cables USB y placas de desarrollo. Los componentes de software consisten en los controladores del sistema operativo y un programa escrito en el lenguaje C++, que recibe y envía las tramas de datos hacia y desde la WSN a través del puerto USB. La arquitectura del sistema de interconexión para el laboratorio remoto tradicional se muestra en la figura 2. Esta arquitectura se basó en el modelo de laboratorio remoto destinado a docencia presentado en [24].

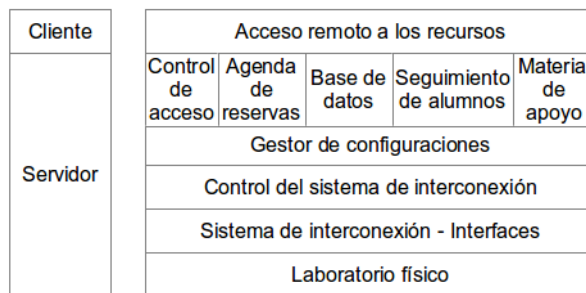


Figura 1: Modelo de capas de Marianetti para laboratorios remotos tradicionales

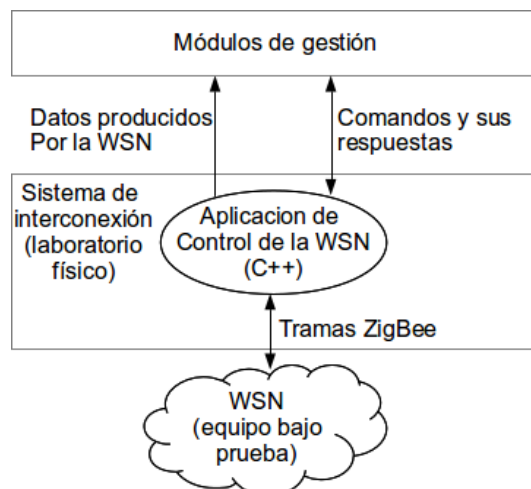


Figura 2: sistema de interconexión del prototipo de laboratorio remoto tradicional

4.2. Implementación Basada en el Cloud

Algunas de las ventajas del Cloud para implementar laboratorios remotos son:

- Alta disponibilidad y robustez de los módulos implementados en el Cloud.
- Alta escalabilidad.
- Alta capacidad de procesamiento y almacenamiento de datos.

Por estas razones, se decidió implementar el laboratorio remoto en el Cloud. Esta implementación se basó en el modelo de capas de Marianetti. Con respecto a un laboratorio tradicional, el sistema de interconexión es el único componente que requiere modificaciones importantes para implementar el laboratorio remoto en el Cloud.

Para poder desplegar los módulos de gestión del laboratorio remoto en el Cloud, el sistema de interconexión debe interconectar el sistema bajo prueba con los módulos de gestión implementados en el Cloud. Para lograr este objetivo se empleó una aplicación cliente-servidor que permite conectar el equipo a prueba con los módulos de gestión desplegados en el Cloud. En consecuencia, el sistema de interconexión se implementó en dos sitios diferentes: (1) el Cloud, y (2) el laboratorio físico, como se muestra en la figura 3. Esta aplicación se implementó empleando lenguaje de programación Java.

Para controlar la WSN, se implementó un programa, denominado aplicación de control de la WSN (figura 3), que se ejecuta en una PC instalada en el laboratorio en el cual se encuentra desplegada la WSN, la cual se comunica con la misma a través de un cable USB. Este programa codifica y decodifica tramas de datos enviadas hacia y desde los nodos. La aplicación de control de la WSN se comunica con la aplicación cliente del sistema de interconexión mencionada anteriormente.

Los nodos se encuentran desplegados en un laboratorio de la Universidad Nacional de Cuyo, en Mendoza. El nodo coordinador se conecta con la PC que alberga la aplicación de control de la WSN a través de un cable USB, alimentándose a través de dicho cable. Los demás nodos se alimentan a través de la línea de energía eléctrica mediante transformadores, y se comunican con el nodo coordinador inalámbricamente.

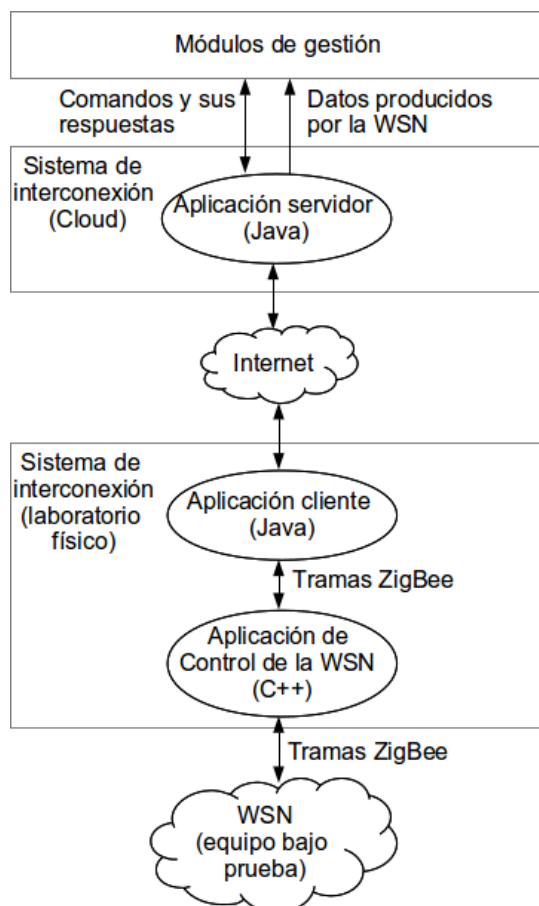


Figura 3: Sistema de interconexión del prototipo de laboratorio remoto basado en el Cloud

4.3. Proveedor de Servicios de Cloud Computing

El sistema implementado utiliza servicios de Cloud Computing. Existen varios proveedores de servicios de Cloud Computing, y es necesario establecer un criterio de selección. El proveedor de servicios de Cloud Computing tiene que proporcionar al menos las siguientes características:

- Almacenamiento de datos en el Cloud.
- Soporte para la comunicación con una aplicación remota. Esta aplicación remota es la encargada de controlar el equipo bajo prueba. La misma debe ser capaz de acceder a los puertos USB y al sistema operativo del ordenador remoto.
- Servicios de fecha y hora.

Además, el siguiente requisito no es necesario, pero es deseable:

- Un mecanismo de autenticación de usuarios proporcionado por el proveedor de servicios de Cloud Computing.

Los tres primeros requisitos citados arriba son necesarios para implementar el laboratorio remoto. El proveedor de servicios de Cloud Computing debe permitir almacenar

datos y consultar la fecha y la hora para poder implementar el sistema de seguimiento de usuarios en el Cloud. De esta manera, el almacenamiento de datos será seguro e independiente del laboratorio físico, por lo que puede funcionar incluso si el laboratorio físico está fuera de servicio. La comunicación con una aplicación remota es necesaria para permitir implementar el sistema de interconexión con el equipo bajo prueba.

En cuanto al servicio de autenticación de usuarios, es preferible que el mismo sea provisto por el proveedor de servicios de Cloud.

Las características requeridas del proveedor de servicios de Cloud Computing son básicas, ya que no se necesita potencia de procesamiento alta o gran espacio para almacenamiento de datos. Por esta razón, se espera que la mayoría de los proveedores de servicios de plataforma (PaaS) o servicios de infraestructura (IaaS) pueda cumplir con estos requisitos. Para esta primera aplicación, el proveedor de servicios de Cloud Computing elegido fue Google App Engine (GAE) [26], que proporciona servicios de plataforma (PaaS). Estos servicios permiten a los usuarios crear aplicaciones que se ejecuten en la infraestructura de Google. En trabajos futuros se estudiará la utilización de otros proveedores de servicios de Cloud Computing.

5. Aplicación a Actividades de Formación

Con base en el análisis presentado en la sección 3, es posible conocer las características que debe cumplir un laboratorio remoto destinado a tareas de formación y entrenamiento de usuarios. A partir de esta información, se propuso un prototipo de laboratorio remoto de WSN destinado a actividades de formación de usuarios. En esta sección se describen las características del prototipo propuesto.

5.1. Interfaz Web

Como se menciona en [25], la interfaz web permite a los usuarios interactuar con el laboratorio remoto para realizar las siguientes tareas:

- Reservar un intervalo de tiempo para acceder al laboratorio remoto.
- Cambiar y monitorear los valores de los parámetros de configuración de los nodos.
- Permitir a los usuarios supervisores controlar la actividad realizada por otros usuarios.

La interfaz web debe ser didáctica, fácil de usar e intuitiva. Para construir la interfaz web, se han seleccionado un conjunto de parámetros de configuración de los nodos bajo prueba. Para cada uno de estos parámetros, se han seleccionado un conjunto de valores predefinidos. Estos parámetros son los mismos que los

usuarios programarían si los nodos estuvieran conectados a su ordenador. Por ejemplo, para los nodos XBee, si el usuario desea programar la potencia de transmisión, debe configurar los valores de los parámetros PM (power mode) y PL (power level), y si el usuario quiere configurar los valores de la frecuencia de reporte de datos, debe configurar los parámetros ST (tiempo antes de entrar en modo bajo consumo), SP (tiempo en modo bajo consumo) y SN (número de ciclos en modo bajo consumo) [27].

La figura 4 muestra una captura de pantalla de parte de la interfaz web. Para cada nodo hay una lista de parámetros de configuración. Cada cuadro que contiene un parámetro en la figura 4 es un botón, sobre el cual el usuario puede hacer clic y acceder a una lista desplegable de posibles valores u opciones de configuración. En el caso de los pines de entrada/salida, el usuario puede configurar su función (deshabilitado, entrada digital, entrada analógica o salida digital) y visualizar el valor leído por cada pin. La interfaz web ha sido desarrollada utilizando lenguajes de programación HTML y JavaScript. Estos lenguajes son aceptados por la mayoría de los navegadores de Internet más populares tanto para ordenadores como para teléfonos móviles.

La interfaz web mostrada en la figura 4 se implementó usando Google Web Toolkit (GWT), una plataforma para la creación de servicios web que se ejecutan en un servidor y aplicaciones cliente que invocan estos servicios [26]. Estos servicios se utilizan para implementar la comunicación entre los usuarios y el laboratorio remoto. Si se compara la interface web de este laboratorio remoto basado en el Cloud con la interface web del laboratorio remoto tradicional presentado en [25], puede verse que la misma no sufre modificaciones importantes. Por lo tanto se concluye que la infraestructura subyacente (Cloud o servidor tradicional) no afecta al funcionamiento de la interface web.

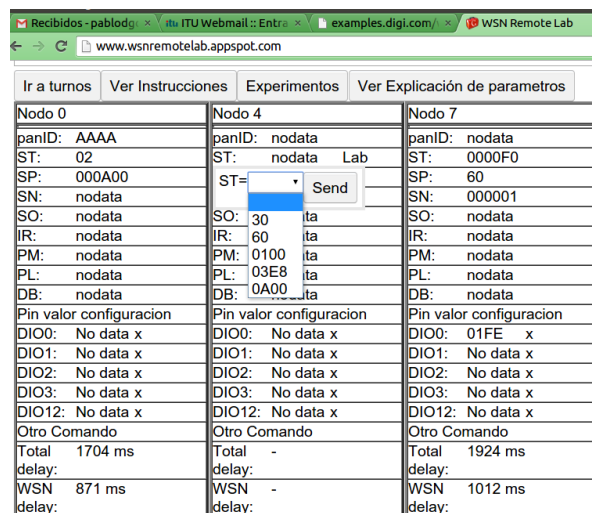


Figura 4: Configuración de parámetros a través de la interfaz web.

5.2. Módulo de registro de actividad de los usuarios

Como se mencionó en la tabla 1 (sección 3), una característica deseable de un laboratorio remoto destinado a tareas de formación es que almacene la actividad realizada por los usuarios, para que usuarios entrenadores o profesores puedan supervisar dicha actividad.

En el laboratorio remoto presentado en este artículo, cada acción realizada por los usuarios se almacena en un registro. En cada registro se almacena la fecha y la hora, el nombre de usuario y la acción realizada. Los registros se almacenan en la base de datos de GAE.

La implementación actual del laboratorio remoto utiliza una implementación del estándar JDO (Java Data Objects) proporcionado por GAE. Este estándar permite almacenar objetos de Java en una base de datos. Cada registro de actividad de usuarios se implementa mediante un objeto de Java con tres variables correspondientes a los datos mencionados anteriormente (el usuario, la fecha y hora en la cual realiza una acción, y la acción realizada), y procedimientos que permiten acceder y manipular esos datos.

Al igual que en [25], la información almacenada por estos registros se presenta en forma de una lista ordenada de datos. Para trabajos futuros se propone implementar interfaces para presentar esta información de manera que resulte más simple visualizar y analizar esta información.

5.3. Otros Módulos que Componen el Laboratorio Remoto de WSN

Las principales contribuciones de este trabajo son la implementación basada en el Cloud, la implementación de una interface web de fácil uso e intuitiva y la implementación de un registro de la actividad realizada por los usuarios, las cuales son características útiles para emplear este laboratorio remoto en la formación de usuarios, pensando en futuras aplicaciones en docencia. Sin embargo, un laboratorio remoto requiere otros módulos. Los mismos se han construido de modo similar a los laboratorios remotos que se mencionan en la sección 2. Estos módulos no se describen en detalle en este trabajo ya que su implementación no constituye un aporte novedoso. A continuación se resumen brevemente sus características.

El control de acceso se realiza utilizando el servicio de autenticación de usuarios proporcionado por GAE. Este servicio utiliza la cuenta de usuario y contraseña de Gmail para poder autenticar a los usuarios. Para poder acceder al laboratorio remoto, los usuarios deben primero acceder a su cuenta de Gmail a través de la interfaz web de dicho servicio de correo electrónico.

El sistema de reserva de turnos es una aplicación implementada en lenguaje Java que se ejecuta en la infraestructura de GAE. Todos los usuarios que tienen una

cuenta de Gmail pueden tener acceso al laboratorio remoto y reservar un turno para su utilización. Los turnos se implementan mediante registros que se almacenan de manera ordenada en la base de datos.

Como se mencionó en la sección 5.2, se emplea el estándar JDO para implementar la base de datos. La misma almacena toda la información necesaria para el funcionamiento del laboratorio remoto, como la actividad realizada por los usuarios, agenda de turnos, configuraciones de los nodos, etc. Esta base de datos almacena objetos de Java. Para cada tipo de datos que se almacena se ha creado una clase, de la cual se crean objetos por cada registro de datos que se necesita almacenar.

Todos los componentes de software fueron implementados en diferentes lenguajes (c++, Java, JavaScript y HTML) empleando el entorno de desarrollo Eclipse. Una descripción más detallada del funcionamiento de estos módulos se presenta en [25].

6. Experimentos

Esta sección tiene el objetivo de mostrar algunos de los posibles experimentos que los usuarios pueden realizar empleando el laboratorio remoto presentado en este trabajo. Los experimentos presentados en esta sección fueron propuestos en [25]. En esta sección se presentan resultados y conclusiones de la realización de los mismos.

6.1. Tiempo de Respuesta de la WSN

Con este experimento, el usuario puede ver cómo la configuración de un nodo afecta a su tiempo de respuesta. El usuario puede programar el tiempo de respuesta de los nodos mediante la configuración de los parámetros de los nodos XBee denominados ST, SP y SN, que tienen las siguientes funciones:

- ST: Intervalo de tiempo de inactividad que debe transcurrir antes de que el nodo pase al modo de bajo consumo.
- SP: Intervalo de tiempo que el nodo permanece en modo de bajo consumo.
- NS: Número de ciclos de SP durante la cual el nodo permanece en modo de bajo consumo.

La figura 5 muestra cómo el ciclo de trabajo de un nodo sensor XBee se configura a través de los parámetros ST, SP y SN. El nodo puede estar en dos estados: (1) activo o (2) bajo consumo (denominado *sleep* en la literatura). La figura 5 muestra tres intervalos de tiempo diferentes, que son:

- Nodo activo realizando tareas: Cuando el nodo pasa de un estado de bajo consumo de energía y al estado activo, trata de comunicarse con el nodo coordinador para recibir los mensajes destinados a él, sensa sus pines de entrada/salida según su configuración, procesa los mensajes que haya enviado el nodo coordinador, y envía al

coordinador la información necesaria (incluyendo las lecturas de sus pines).

- **Nodo activo no realizando ninguna tarea:** Una vez que el nodo ha terminado de realizar todas sus tareas, espera un intervalo de tiempo *ST* por nuevas comunicaciones con el nodo coordinador. Si no hay ningún mensaje desde el nodo coordinador durante el intervalo de tiempo *ST*, el nodo cambia al modo de bajo consumo de energía.
- **Nodo en el estado de bajo consumo de energía:** el nodo permanece en modo de bajo consumo un intervalo de tiempo *SNxSP* antes de volver al estado activo.

Cuando el usuario envía un comando a un nodo, este comando puede alcanzar el nodo coordinador en cualquier momento del ciclo de trabajo. Si el nodo de destino está en estado activo, el nodo coordinador enviará el mensaje al mismo inmediatamente. Pero si el nodo destino se encuentra en estado bajo consumo, el nodo coordinador espera hasta que el nodo destino se despierte para enviarle su mensaje. Como resultado, el tiempo de respuesta puede variar entre un valor mínimo que depende de la velocidad de respuesta del protocolo ZigBee (del orden de 100 milisegundos), y un valor máximo en función de los valores dados para los parámetros de *ST*, *SN* y *SP*.

Con el fin de conocer el tiempo de respuesta y la latencia total, el usuario envía un comando a cualquier nodo, y cuando la respuesta vuelve, la interfaz web muestra el tiempo de respuesta de la WSN.

Para llevar a cabo este experimento se utilizó el comando DB. Este comando solicita el valor del indicador RSSI (intensidad de señal recibida) [27]. La tabla 2 muestra las mediciones de tiempo para diferentes configuraciones de los parámetros de *ST*, *SP* y *SN*. Para realizar el experimento, se utiliza un nodo cualquiera de la WSN y el nodo coordinador. Se realizó una serie de 30 mediciones para cada combinación de parámetros. La tabla 2 muestra el valor medio, el valor más pequeño, el valor más grande y la desviación estándar para cada conjunto de 30 mediciones para cada combinación de parámetros.

Se puede observar que, para cada combinación de parámetros, el tiempo de respuesta más corto no depende de los valores de los parámetros, mientras que el tiempo de respuesta más alto, el promedio y desviación estándar de los tiempos de respuesta dependen de los valores dados a *ST*, *SP* y *SN*. Las filas 1 y 2 muestran que si el intervalo de tiempo en estado activo (dependiente de *ST*) aumenta, el tiempo medio de respuesta disminuye. Esto se debe a que al aumentar el intervalo de tiempo durante el cual el nodo de destino se mantiene activo, aumenta la probabilidad de que la petición llegue a la WSN cuando el nodo de destino está esperando nuevos mensajes, y por lo tanto puede responder inmediatamente. Por otro lado, al aumentar el intervalo de tiempo en bajo consumo de energía (dependiente de *SP* y *SN*), el tiempo de respuesta más corto no aumenta, pero el tiempo medio de respuesta, el tiempo de respuesta más largo y la desviación estándar aumentan.

Para llevar a cabo este experimento cuando un laboratorio remoto no es accesible, el usuario debe: (1) desplegar una WSN y configurar los nodos con firmware adecuado de acuerdo con su función, (2) desarrollar un programa de comunicación con el nodo coordinador a través del puerto USB, (3) enviar tramas de datos o comandos a los nodos para realizar las operaciones deseadas, (4) implementar un mecanismo para medir intervalos de tiempo del orden de los milisegundos y (5) interpretar las tramas recibidas por el coordinador para analizar los resultados. Por lo tanto, el laboratorio remoto, además de permitir al usuario experimentar con hardware real, facilita la ejecución de los experimentos. El usuario no necesita desplegar una WSN, configurar los nodos a partir de cero, ni escribir programas para comunicarse con nodos. El usuario sólo tiene que seleccionar los valores deseados para los parámetros e interpretar los resultados.

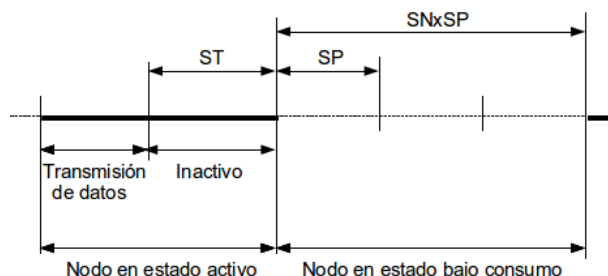


Figura 5: Ciclo de trabajo de un nodo XBee

Tabla 2: Tiempo de respuesta de la WSN

Valor de los parámetros			Tiempo de respuesta de la WSN			
			Promedio	Menor valor	Mayor valor	Desviación estándar
ST	SP	SN				
30	30	1	827 ms	465 ms	1.30 s	244 ms
3E8	30	1	667 ms	349 ms	869 ms	227 ms
3E8	3E8	1	4.43 s	776 ms	10.2 s	3.27 s
3E8	3E8	3	10.1 s	630 ms	20.3 s	8.10 s

6.2. Potencia de RF Recibida

La potencia de radiofrecuencia (RF) recibida por un nodo depende de la potencia de RF transmitida por el nodo transmisor y la atenuación de la señal. La atenuación depende de muchos factores, incluyendo la distancia entre los nodos, la presencia de objetos, la altura de los nodos con respecto al plano de tierra, la posición de las antenas, etc. La mayor parte de los nodos disponibles comercialmente proporcionan un mecanismo para medir la potencia de RF de la señal recibida. Esto permite a los programadores ajustar la potencia de transmisión al valor requerido.

Los nodos XBee permiten conocer el valor RSSI (indicador de intensidad de la señal recibida), que indica

la potencia de RF recibida por un nodo en dBm. Este valor de RSSI se corresponde con el último paquete de datos recibido por el nodo (que puede ser un paquete de datos, una respuesta, etc.). Además, dos parámetros permiten ajustar la potencia transmitida y la sensibilidad del receptor. Por un lado, el comando PL (nivel de potencia) permite ajustar la potencia de transmisión de un nodo pudiendo elegir entre 5 valores, tal como se muestra en la tabla 3. Por otra parte, cuando el parámetro de PM es igual a 1, aumenta la potencia de transmisión por 2 dB en comparación con los valores mostrados en la tabla 3 y la sensibilidad del receptor en 1 dB.

Con este experimento, el usuario puede observar la variación de la potencia de RF recibida por un nodo en función de la potencia de RF transmitida por el nodo transmisor y la distancia entre los nodos.

Para llevar a cabo este experimento, el usuario debe utilizar el comando DB (sección 5.2). La metodología utilizada en este experimento consiste en modificar la potencia de transmisión del nodo coordinador, y verificar el valor de RSSI recibido por el resto de los nodos. El nodo 1 es el más cercano al nodo coordinador, luego el nodo 2, y así sucesivamente. La tabla 4 muestra los resultados de este experimento con algunos nodos de la WSN que forman el laboratorio remoto. Los valores de RSSI se expresan en dBm (como se explica en la hoja de datos de los nodos XBee [28]). Se puede observar que, cuanto mayor es la potencia de transmisión o menor es la distancia entre los nodos, mayor es la potencia de RF recibida por el nodo de destino.

La realización de este experimento requiere que el usuario posea conocimientos acerca de la configuración de los nodos de una WSN y sobre transmisión de señales de RF. Sin embargo, los usuarios no necesitan desplegar una WSN, programar los nodos, o interpretar las tramas de datos, debido a que estas tareas son realizadas por el laboratorio remoto. En [29] se describe el procedimiento necesario para realizar un experimento como este si no se cuenta con un laboratorio remoto.

6.3. Tiempo de Respuesta en Función de la Ocupación del Canal de Comunicaciones

Los nodos XBee utilizados en este laboratorio remoto implementan el estándar IEEE 802.15.4 [28] en sus capas física y MAC. Este utiliza el protocolo CSMA/CA para acceder y compartir el canal de comunicaciones inalámbrico. De acuerdo con lo establecido por la norma IEEE 802.15.4, cuando un nodo necesita transmitir datos, primero verifica varias veces si el canal de comunicaciones está libre, añadiendo intervalos de tiempo aleatorios entre cada verificación. Si el canal de comunicaciones está libre para todas las verificaciones, el nodo transmite sus datos. Si el canal de comunicación está ocupado en cualquiera de las verificaciones, el nodo espera un tiempo aleatorio y reinicia el proceso de

verificación. Este mecanismo está destinado a reducir la probabilidad de colisiones.

Cuanto mayor es el número de nodos o mayor la frecuencia de transmisión de datos, mayor es la probabilidad de que un nodo encuentre el canal de comunicaciones ocupado y tenga que esperar para enviar datos. Como resultado, el tiempo medio de respuesta aumenta. Este experimento tiene como objetivo medir el tiempo de respuesta promedio de un nodo en función de la frecuencia de transmisión de los otros nodos en la red.

Tabla 3: Potencia transmitida por un nodo en función del parámetro PL (de acuerdo con la hoja de datos de los nodos XBee [27])

PL value	Output RF power
0	-7 dBm
1	-3 dBm
2	-1 dBm
3	+1 dBm
4	+3 dBm

Tabla 4: Valor de RSSI (en dBm) recibido por un nodo para diferentes potencias de transmisión del nodo coordinador y diferentes distancias

Coordinador config.		RSSI recibido por diferentes nodos			
PM	PL	Nodo 1	Nodo 2	Nodo 6	Nodo 7
00	00	-30 dBm	-43 dBm	-67 dBm	-89 dBm
01	00	-28 dBm	-37 dBm	-61 dBm	-85 dBm
00	01	-28 dBm	-34 dBm	-61 dBm	-84 dBm
01	01	-27 dBm	-31 dBm	-57 dBm	-76 dBm

Con el fin de realizar el experimento, el nodo 1 se utiliza como nodo bajo prueba. Los otros nodos (del 2 al 10) se han configurado para transmitir datos de forma periódica. Se midió el tiempo de respuesta del nodo 1, para diferentes tasas de transmisión de los demás nodos.

Los nodos 2 al 10 están configurados para transmitir periódicamente datos de manera que, en conjunto, envíen al canal de comunicaciones inalámbrico 0, 10, 20, 30 o 40 paquetes de datos por segundo. El valor de ocupación del canal de comunicaciones igual a 0 corresponde a la situación en que ningún nodo transmite datos, es decir, el canal de comunicaciones está desocupado, y solo transmite datos el nodo bajo prueba (nodo 1). Es un valor de referencia o situación ideal, y el tiempo de respuesta correspondiente es el tiempo de respuesta que habría si los nodos pudieran transmitir sin interferirse entre sí. Para cada uno de estos valores de ocupación del canal de comunicaciones, se tomaron 30 mediciones del tiempo de

respuesta del nodo 1. Se calcularon el valor medio y la desviación estándar. La tabla 5 muestra los resultados.

Se puede observar que, cuanto mayor la ocupación del canal de comunicaciones, es mayor el tiempo de respuesta promedio de un nodo a un comando. Puede notarse que para ocupaciones del canal de comunicaciones mayores a 30 paquetes de datos por segundo, el tiempo de respuesta promedio y su desviación estándar aumentan considerablemente. Sin embargo, para ocupaciones del canal de comunicaciones menores a 20 paquetes por segundo, el tiempo medio de respuesta del nodo 1 es cercano al valor ideal (0.6 segundos) y no varía considerablemente. Este resultado se puede atribuir a que, cuanto mayor es la ocupación del canal de comunicaciones, mayor es la probabilidad de que un nodo encuentre el canal de comunicaciones ocupado y tenga que esperar para poder transmitir datos.

Es necesario aclarar que, si bien la ocupación del canal de comunicaciones afecta significativamente el tiempo de respuesta de un nodo, dicho tiempo de respuesta depende también de otros parámetros, como la configuración del propio nodo bajo prueba y las configuraciones de los ciclos de trabajo de todos los nodos de la WSN. El laboratorio remoto presentado en este trabajo y la metodología presentada para realizar este experimento de prueba de concepto se utilizaron para realizar mediciones más detalladas, en función de un mayor número de valores de ocupación del canal de comunicaciones y para diferentes configuraciones de los nodos de la WSN. Los resultados obtenidos fueron publicados en otro artículo [30]. Este resultado muestra que el laboratorio remoto y la metodología propuesta para realizar experimentos de prueba de concepto también pueden emplearse para realizar experimentos científicos.

Tabla 5: Tiempo de respuesta promedio para el nodo 1 (segundos) en función de la ocupación del canal de comunicaciones.

Ocupación del canal de comunicaciones (paquetes por segundo)	Tiempo de respuesta promedio	Desviación estándar
0	0.6 s	0.58 s
10	0.8 s	0.75 s
20	0.7 s	0.8 s
30	3.1 s	2.8 s
40	8.0 s	6.2 s

7. Conclusiones y Trabajos Futuros

Este artículo presenta un laboratorio remoto dirigido a la formación de usuarios desplegado en una infraestructura de Cloud Computing. Se ha construido un prototipo y se diseñaron, realizaron y analizaron diferentes experimentos

de validación. Estos experimentos se pueden ejecutar rápidamente, ya que el usuario no necesita desplegar una WSN ni un sistema de adquisición de datos. Los usuarios pueden poner toda su atención en el concepto que necesitan desarrollar o entrenar, sin requerir tiempo adicional en la configuración del experimento.

Los experimentos presentados en este trabajo ponen a prueba conceptos fundamentales sobre WSN que son importantes en aplicaciones reales, por ejemplo:

- Tiempo de respuesta de nodos WSN bajo diferentes condiciones de trabajo.
- Intensidad de señal recibida en función de la potencia de transmisión y la distancia.
- Congestión en la WSN bajo diferentes condiciones de trabajo.

Esto demuestra que los laboratorios remotos pueden ser una herramienta poderosa para realizar actividades de formación de usuarios, útil tanto para usuarios principiantes, usuarios avanzados o estudiantes.

El trabajo futuro incluye:

- Realizar experimentos de rendimiento, para conocer las limitaciones y deficiencias del prototipo, y avanzar hacia un producto final.
- Probar el laboratorio remoto con estudiantes. Esto permitirá conocer sus opiniones y mejorar el diseño del laboratorio remoto.
- Probar el laboratorio remoto con otros tipos de sistema bajo prueba.
- Proponer otros experimentos predefinidos, que permitan poner a prueba otros conceptos relacionados con las WSN o la transmisión de señales inalámbricas (nodos a diferentes alturas, diferentes obstáculos, etc).
- Implementar una interface para presentar la información de la actividad realizada por los usuarios de manera que sea simple de interpretar.

En este momento el laboratorio remoto está siendo preparado para emplearlo en el entrenamiento de un grupo de estudiantes que participan en un proyecto de investigación que empleará el uso de WSN para predecir heladas. Se espera que estos estudiantes aprendan los conceptos básicos de WSN y de programación de los nodos, de modo que puedan configurar en campo los nodos de una WSN y realizar el mantenimiento necesario.

Referencias

- [1] M. Waldrop, "Campus 2.0," *Nature*, vol. 495, no. 7440, pp. 160–163, 2013.
- [2] M. Waldrop, "Education online: the virtual lab." *Nature*, vol. 499, no. 7458, pp. 268–270, 2013.

- [3] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292 – 2330, 2008.
- [4] E. Gilbert, "Research issues in wireless sensor network applications: a survey," *International Journal of Information and Electronics Engineering*, vol. 2, no. 5, pp. 702–706, 11 2012.
- [5] G. Zhao, "Wireless sensor networks for industrial process monitoring and control: A survey," *Network Protocols and Algorithms*, vol. 3, no. 1, pp. 46–63, 2011.
- [6] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599 – 616, 2009.
- [7] P. Mell and T. Grance, "The NIST definition of Cloud Computing (Draft)," *NIST Special Publication*, vol. 800, p. 145, 2011.
- [8] A. Maiti and B. Tripathy, "Remote laboratories: Design of experiments and their web implementation." *Journal of Educational Technology & Society*, vol. 16, no. 3, 2013.
- [9] L. Steyn and G. Hancke, "A survey of wireless sensor network testbeds," in *AFRICON*, 2011, Sept 2011, pp. 1–6.
- [10] K. El-Darymli and M. Ahmed, *Wireless Sensor Networks and Energy Efficiency: Protocols, Routing and Management*. IGI Global, 2012, ch. *Wireless Sensor Network Testbeds*, pp. 148–205.
- [11] H. Hellbruck, M. Pagel, A. Kroller, D. Bimschas, D. Pfisterer, and S. Fischer, "Using and operating wireless sensor network testbeds with WISEBED," in *Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2011 The 10th IFIP Annual Mediterranean, june 2011, pp. 171 –178.
- [12] P. Godoy, L. Iacono, R. Cayssials, and C. García Garino, "A survey of WSN testbeds deployment," *Congreso Argentino de Sistemas Embebidos (CASE)*, 2012.
- [13] P. Orduna, A. Gomez-Goiri, L. Rodriguez-Gil, J. Diego, D. Lopez-de Ipina, and J. Garcia-Zubia, "wCloud: Automatic generation of WebLab-Deusto deployments in the Cloud," in *Remote Engineering and Virtual Instrumentation (REV)*, 2015 12th International Conference on, Feb 2015, pp. 223–229.
- [14] M. Tawfik, C. Salzmann, D. Gillet, D. Lowe, H. Saliyah-Hassane, E. Sancristobal, and M. Castro, "Laboratory as a service (LaaS): A model for developing and implementing remote laboratories as modular components," in *Remote Engineering and Virtual Instrumentation (REV)*, 2014 11th International Conference on, Feb 2014, pp. 11–20.
- [15] S. Misra, S. Chatterjee, and M. Obaidat, "On theoretical modeling of sensor Cloud: A paradigm shift from wireless sensor network," *Systems Journal*, IEEE, vol. PP, no. 99, pp. 1–10, 2014.
- [16] L. Iacono, C. García Garino, O. Marianetti, and C. Párraga, "Wireless sensor networks: A software as a service approach," *Prospective and Ongoing Projects, VI Latin American Symposium on High Performance Computing (HPCLatAm 2013)*, Mendoza, Argentina, pp. 184–195, 2013.
- [17] L. Iacono, P. Godoy, O. Marianetti, C. García Garino, and C. Párraga, "Estudio de la integración entre WSN y redes TCP/IP," *Memoria de Trabajos de Difusión Científica y Técnica* 10, 2012, ISSN 1510-7450.
- [18] M. W. Bazzaza; K. Salah: *Using the Cloud to Teach Computer Networks*. In: 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC), pp. 310–314.
- [19] K. Salah: *Harnessing the Cloud for Teaching cybersecurity*. In: *Proceedings of the 45th ACM Technical Symposium on Computer Science Education, SIGCSE '14*, ACM, New York, NY, USA, ISBN 978-1-4503-2605-6, pp. 529–534,
- [20] K. Salah; M. Hammoud; S. Zeadally: *Teaching Cybersecurity Using the Cloud*. *IEEE Transactions on Learning Technologies*, 2015, vol. 8(4), pp. 383–392, ISSN 1939-1382.
- [21] P. Godoy, *Plataforma de Desarrollo de Laboratorios Remotos de Redes de Sensores Inalámbricos basados en Cloud Computing*. Universidad de Mendoza, Facultad de Ingeniería, Tesis de Doctorado en Ingeniería, 2016.
- [22] G. Werner-Allen, P. Swieskowski, and M. Welsh, "MoteLab: a wireless sensor network testbed," in *Information Processing in Sensor Networks*, 2005. *IPSN 2005. Fourth International Symposium on*, april 2005, pp. 483 – 488.
- [23] O. Marianetti, *Laboratorios remotos, un aporte para su diseño y gestión*. Universidad de Mendoza, Facultad de Ingeniería, Tesis de Maestría en Teleinformática, 2006.
- [24] K. Jona and D. Uttal, "Don't forget the teacher: New tools to support broader adoption of remote labs," in *Remote Engineering and Virtual Instrumentation (REV)*, 2013 10th International Conference on, Feb 2013, pp. 1–2.
- [25] P. Godoy, R. Cayssials and C. García Garino, "A WSN Testbed for Teaching Purposes," *IEEE Latin America Transactions*, vol. 14, issue 7, 2016.
- [26] Google Inc. (2013) Google App Engine website. Retrieved in 2015.
- [27] Digi International Inc. (2008) XBee ZB low power ZigBee module with integrated wire antenna datasheet. Retrieved in 2015.
- [28] Institute of Electrical and Electronic Engineers, "IEEE std 802.15.4-2003," pp. 1–670, 2003.
- [29] P. Godoy, L. Iacono, R. Cayssials, C. Párraga, C. García Garino, "Effect of working conditions over the

performance in ZigBee WSN”, In Capítulo Sistemas de Control, Anales de la primera edición del Congreso de la Sección Argentina de IEEE (Argencon) 2012, Córdoba, Argentina, ISBN: 987-572-076-3

[30] P. Godoy, R. Cayssials, and C. García Garino, “Zigbee WSN Round Trip Latency in Function of Channel Occupation and Nodes Configuration”, In IEEE ArgenCon 2016.

Dirección de Contacto de los Autores:

Pablo Godoy
Mendoza
Argentina
e-mail: pgodoy@itu.uncu.edu.ar
sitio web: <http://itic.uncu.edu.ar/>

Ricardo Cayssials
Buenos Aires
Argentina
e-mail: rcayssials@frbb.utn.edu.ar

Carlos García Garino
Mendoza
Argentina
e-mail: cgarcia@itu.uncu.edu.ar
sitio web: <http://itic.uncu.edu.ar/>

Pablo Godoy es Doctor en Ingeniería desde 2016 (Universidad de Mendoza) e Ingeniero Electrónico desde 2008 (UTN-FRM), y se desempeña como docente investigador en la Universidad Nacional de Cuyo.

Ricardo Cayssials es Doctor en Ingeniería desde 1999 (Universidad Nacional del Sur) e Ingeniero Electrónico desde 1993 (Universidad Nacional del Sur), y se desempeña como docente investigador en la Universidad Tecnológica Nacional, Facultad Regional Bahía Blanca.

Carlos García Garino es director el ITIC y profesor titular de la facultad de Ingeniería de la Universidad Nacional de Cuyo. Se graduó de Ingeniero Civil en 1978 (UBA) y de Doctor Ingeniero en 1993 (Universidad Politécnica de Cataluña).
