

# Control Dómótico por medio de Dispositivos Móviles Celulares basado en una Arquitectura Distribuida

C. Martínez, S. Pino y S. Castillo

**Resumen**— Este artículo presenta una arquitectura distribuida de transmisión de datos, la cual es base para el desarrollo de un prototipo software domótico para el control de dispositivos de una casa a larga distancia. La arquitectura presentada está compuesta de diversos dispositivos físicos que se encuentran separados geográficamente.

El prototipo software desarrollado consta de dos aplicaciones móviles celulares y dos servicios web, completando el software requerido por la arquitectura distribuida, y una simulación sobre la cual se refleja el control de los dispositivos.

Para la transmisión de diversos mensajes se utiliza HTTP, SMS [1] y Bluetooth [2]-[4] según sea el caso.

Se ha podido implementar el prototipo en dispositivos reales convencionales de moderado costo (celulares gama media) pudiendo controlar (activar/desactivar) satisfactoriamente dispositivos simulados, probando la funcionalidad de la arquitectura distribuida.

**Palabras clave**— Bluetooth, casas inteligentes, dispositivos móviles, domótica (*Smart Homes*), Java, mensajes HTTP, mensajes SMS (*Short Message Service*).

## I. INTRODUCCIÓN

La preservación de los recursos naturales y la búsqueda de fuentes energéticas alternativas son materias de constante investigación actualmente. Mientras se logran importantes avances en materia de energías alternativas, un gran esfuerzo se centra en disminuir el consumo energético actual. Países como Japón y España empiezan a hacer frente a esta realidad, y entre sus aportes se apunta a la masificación de los sistemas domóticos [5].

Los sistemas domóticos, en su definición más básica, contribuyen con el ahorro energético de las casas donde son implementados, pero estos sistemas van más allá; permiten tomar decisiones sobre el entorno del hogar, brindando confort y mejorando la calidad de vida de los usuarios. Los avances tecnológicos abren paso a nuevas formas de interacción. Capacidad de comunicación a grandes distancias, sumado a los avances de los dispositivos móviles amplían las opciones de control de espacios domóticos.

Este documento presenta el desarrollo de un prototipo de sistema domótico con arquitectura distribuida, que permite al

usuario la manipulación de dispositivos electrónicos controlables presentes en una casa simulada, utilizando como panel de acceso su dispositivo móvil de bolsillo.

Para el presente documento, entiéndase por dispositivos electrónico controlables aquellos que han sido adaptados para el control domótico como pueden ser lámparas, puertas, ventanas, etc. Estos dispositivos poseen estado de activación llamado aquí activado/desactivado como palabras genéricas a las diferentes alternativas que pueden ser encendido/apagado, abierto/cerrado, etcétera.

## II. ARQUITECTURA DEL SISTEMA

El sistema domótico consta de cuatro componentes hardware separados geográficamente, los cuales interactúan ordenadamente entre ellos, según corresponda, basándose en la arquitectura cliente/servidor [6].

Los cuatro componentes hardware de la arquitectura distribuida, representada en la Fig. 1 son: servidor web, dispositivo móvil de usuario, dispositivo móvil de la casa y ordenador. Estos componentes y sus interacciones son detallados en las siguientes secciones.

### A. Servidor Web

El servidor web se encarga del mantenimiento de todas las casas y usuarios del sistema, y de los dispositivos electrónicos controlables asociados a cada casa.

Al encargarse del mantenimiento de los datos, brinda soporte a los dispositivos móviles de usuario y de la casa, notificándoles datos importantes referentes a los dispositivos electrónicos como por ejemplo su estado de activación actual.

Las interacciones entre servidor y móvil de usuario, y entre servidor y móvil de la casa se realizan a través de internet por medio de mensajes HTTP.

En la Fig. 1 se puede apreciar la interacción entre el servidor y los dispositivos móviles, mientras hace énfasis en la interacción entre este y la base de datos.

### B. Dispositivo móvil de usuario

El dispositivo móvil de usuario funciona como panel de control de los diferentes dispositivos electrónicos controlables asociados a cada casa.

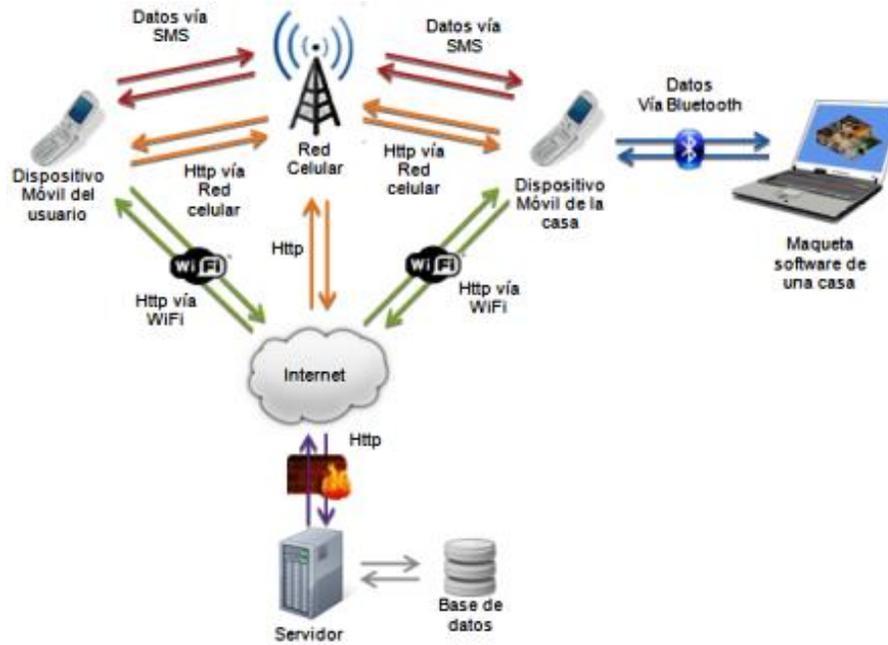


Fig. 1. Arquitectura distribuida.

Por medio de este dispositivo móvil se pueden realizar acciones como la petición de información relativa a los dispositivos electrónicos controlables y la emisión de órdenes de cambio de estado de activación de los mismos.

Durante la petición de información, el dispositivo móvil de usuario interactúa con el servidor web a través de internet utilizando mensajes HTTP.

Durante la emisión de órdenes, este dispositivo móvil interactúa con el dispositivo móvil de la casa a través de las redes de operadores celulares utilizando mensajes SMS.

Se requiere que el dispositivo móvil de usuario posea las capacidades de transmisión y recepción de mensajes SMS, conexión a Internet a través de redes de operadores celulares u otras como WiFi, e interfaz de entrada como panel de teclas o pantalla táctil.

En la Fig. 1 se aprecia la interacción entre el dispositivo móvil del usuario y el servidor web a través de Internet utilizando la red de operador celular o una red WiFi, y la interacción entre los dispositivos móviles de usuario y casa a través de la red de operador celular.

### C. Dispositivo móvil de la casa

Este dispositivo funciona como puente de conexión entre el dispositivo móvil de usuario y el ordenador. En su proceso de interacción recibe una petición enviada por el dispositivo móvil de usuario utilizando un mensaje SMS, intercambia mensajes HTTP con el servidor web durante el proceso de TPV<sup>1</sup> (véase sección IV-A) y según el resultado de este proceso puede llegar a interactuar con el ordenador utilizando mensajes Bluetooth. El dispositivo móvil de la casa finaliza

la interacción transmitiendo un mensaje SMS al dispositivo móvil de usuario. La Fig. 1 muestra la interacción entre el dispositivo móvil de la casa y otros componentes hardware.

En este punto de la arquitectura se pudo haber utilizado un módem GSM/GPRS, pero dado que las funciones de este componente pueden ser satisfechas por un dispositivo móvil celular de gama media a un menor costo, este tipo de dispositivos han sido la elección.

Requerimientos: transmisión y recepción de mensajes SMS, conexión a Internet a través de redes de operadores celulares u otras como WiFi, y transmisión y recepción de datos a través de Bluetooth.

### D. Ordenador

El ordenador recibirá peticiones generadas por el usuario y se encargará de ordenar su ejecución.

El ordenador interactúa con el dispositivo móvil de la casa a través de mensajes Bluetooth.

Se requiere que el ordenador posea la capacidad de transmisión y recepción de mensajes Bluetooth.

## III. DESCRIPCIÓN DEL SISTEMA

Como se puede apreciar, los componentes de la arquitectura descrita anteriormente se encuentran en constante interacción. Cada componente hardware es provisto de al menos un componente software, los cuales se encargan de interpretar y generar los mensajes transferidos a lo largo de la arquitectura.

El sistema distribuido Domolution (contracción de

<sup>1</sup> Third-Party Verification (Verificación por parte de tercero)

Domotic Solution) contiene cinco componentes software: Domolution Web, Domolution Control Core, Domolution Mobile, Domolution Mobile Link y Domolution Execution Core. Estos componentes, al igual que los componentes hardware que los implementan, se encuentran separados geográficamente y en constante comunicación.

#### A. Tipos de dispositivos electrónicos controlables

Domolution maneja dos tipos de dispositivos electrónicos controlables:

- Dispositivos comunes, que por consideración previa pueden ser utilizados por cualquier usuario o miembro de la casa.
- Dispositivos restringidos, que sólo pueden ser utilizados por usuarios con privilegios.

#### B. Estados de un dispositivo electrónico controlable

Un dispositivo electrónico controlable posee dos estados:

- Estado de activación: puede ser activado o desactivado. Son palabras genéricas para estados como encendido/apagado, abierto/cerrado, etcétera.
- Estado de bloqueo: puede ser bloqueado o desbloqueado. El uso de un dispositivo bloqueado dependerá de los privilegios de usuario (véase sección III-C).

#### C. Tipos de usuarios

Se manejan tres tipos de usuarios:

- Usuario técnico o administrador, quien se encarga de las instalaciones de los dispositivos electrónicos controlables en la casa, los registra en el sistema y crea al menos un usuario privilegiado.

- Usuario privilegiado, quien puede manipular todos los dispositivos de la casa. Este usuario puede cambiar el tipo de un dispositivo de común a privilegiado y viceversa, así como cambiar su estado de bloqueo.
- Usuario común, quien puede manipular únicamente dispositivos comunes mientras se encuentran en estado desbloqueado.

#### D. Funcionalidades

Domolution brinda principalmente tres funcionalidades:

- Notificación de dispositivos presentes en una casa y su estado de bloqueo.
- Cambio de estado de los dispositivos electrónicos controlables teniendo en cuenta los privilegios de usuario sobre dicho dispositivo.
- Registrar dispositivos y usuarios en el sistema.

Los subsistemas o componentes software que brindan estas funcionalidades son descritos en la sección III-E.

#### E. Implementación

Como se describe en el inicio de la parte III, Domolution contiene cinco componentes software. Para efectos del desarrollo de este prototipo estos componentes fueron desarrollados utilizando diversas tecnologías Java. Los componentes software se describen en los siguientes apartados.

El diagrama de secuencias de la Fig. 2 muestra los procesos y subsistemas involucrados en la ejecución de una petición generada por el usuario.

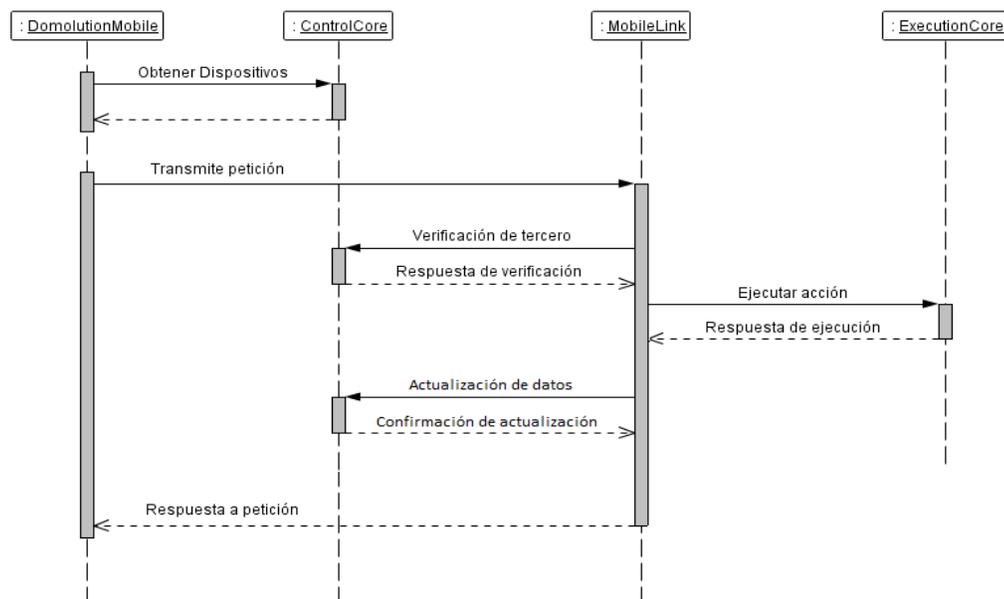


Fig. 2. Subsistemas y procesos involucrados en la ejecución de una petición.

### 1) Domolution Web

Es uno de los servicios alojados en el servidor web.

Permite al usuario técnico registrar en el sistema nuevas casas domóticas y los dispositivos electrónicos controlables instalados en ellas. También le permite a éste la creación de usuarios privilegiados y usuarios comunes.

Así mismo, los usuarios privilegiados pueden hacer uso de éste subsistema para organizar los dispositivos electrónicos de sus casas de la forma más útil para ellos; esta organización e verá reflejada en su interacción con el subsistema integrado en sus dispositivos móviles. Por otra parte, este subsistema brinda a los usuarios privilegiados la posibilidad de administrar a otros usuarios. Los usuarios comunes pueden editar sus datos personales.

Todos los usuarios tienen acceso al historial de modificaciones de dispositivos.

Domolution Web fue desarrollado con la tecnología Java Enterprise Edition [7].

### 2) Domolution Control Core

Este subsistema es uno de los servicios alojados en el servidor web.

Provee servicios de administración de datos a los subsistemas alojados en los dispositivos móviles, lo que supone que se encuentra en constante comunicación con la base de datos central del sistema.

Notifica a Domolution Mobile (véase apartado III-E-3) sobre las casas disponibles para un usuario específico, así como sus dispositivos electrónicos y los estados de los mismos.

Durante una petición generada por un usuario, este subsistema realiza un proceso de TPV (véase sección IV-A) a petición de Domolution Mobile Link (véase apartado III-E-4). En la Fig. 3 se ilustra la interacción entre Domolution Control Core y los dos subsistemas móviles.

Domolution Control Core fue desarrollado en la tecnología Java Enterprise Edition.

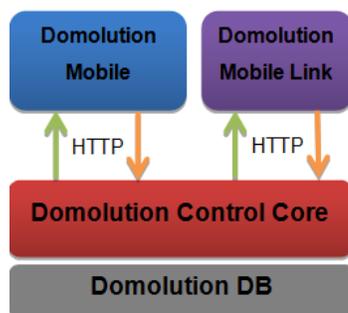


Fig. 3. Esquema de petición/respuesta de Domolution Control Core.

### 3) Domolution Mobile

Domolution Mobile fue desarrollado con la tecnología Java Micro Edition [9], [10], lo que le permite alojarse en dispositivos móviles de usuario.

Su función es permitir al usuario monitorear e interactuar con los dispositivos electrónicos de una casa mediante la generación de peticiones transmitidas como mensajes SMS al sistema Domolution Mobile Link (véase apartado III-E-4).

Por medio de este subsistema, el usuario puede ver detalles de sus casas y dispositivos electrónicos, así como sus estados, gracias a los datos suministrados por Domolution Control Core a través de mensajes HTTP. Estos detalles son los que permiten a un usuario tomar decisiones acerca de los dispositivos de su casa desde cualquier lugar.

El entorno gráfico de éste subsistema está basado en la librería gráfica LWUIT<sup>2</sup> [11], lo que le permite tener una apariencia uniforme en diferentes dispositivos móviles celulares aún en diversas marcas y modelos.

Domolution Mobile está orientado a la conectividad, por lo que permite conexiones a Internet tanto por redes inalámbricas de área local, como WiFi, o por las redes de los operadores celulares.

### 4) Domolution Mobile Link

El subsistema Domolution Mobile Link se ha desarrollado con la tecnología Java Micro Edition y la librería gráfica LWUIT, y reside en el dispositivo móvil de la casa.

Es el subsistema más complejo del sistema Domolution cuando se habla de interacción entre subsistemas. La Fig. 4 muestra la interacción de Domolution Mobile Link con otros subsistemas.

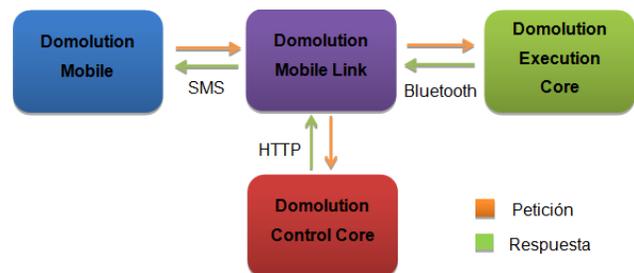


Fig. 4. Interacción de Domolution Mobile Link con otros subsistemas.

Al transmitirse una petición generada por un usuario del subsistema Domolution Mobile, el subsistema Domolution Mobile Link recibe un mensaje SMS del cual deberá extraer la petición, interpretarla y retransmitirla a Domolution Control Core, por medio de un mensaje HTTP, solicitando que ésta sea sometida al proceso de TPV<sup>1</sup> (véase la sección IV-A) en caso de ser una petición de cambio de estado de activación o que ésta sea ejecutada en caso de ser una petición de cambio de estado de bloqueo. La Fig. 5 muestra un diagrama de ejecución de peticiones por parte de Domolution Mobile Link.

<sup>2</sup> Lightweight User Interface Toolkit

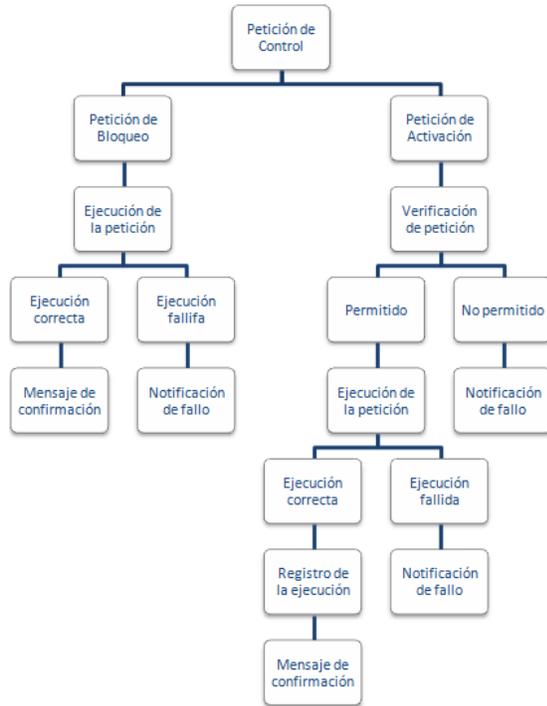


Fig. 5. Ejecución de peticiones por parte de Domolution Mobile Link.

Dado que una petición de cambio de estado de bloqueo no requiere una acción física, al recibir el mensaje HTTP retornado por Domolution Control Core, éste es interpretado, un respectivo código de respuesta es empaquetado en un mensaje SMS y transmitido de regreso a Domolution Mobile, el cual le informa al usuario el resultado de su petición.

Cuando la petición involucra el cambio de estado de un dispositivo, el mensaje HTTP retornado por Domolution Control Core, luego del proceso TPV<sup>1</sup>, es interpretado. Si el mensaje indica que la petición es fallida, un mensaje SMS es retornado a Domolution Mobile con el respectivo código de respuesta. En caso de que el proceso TPV fuera exitoso, Domolution Mobile Link reescribe la petición en un mensaje Bluetooth, el cual transfiere a Domolution Execution Core (véase apartado III-E-5).

Posteriormente se recibe un mensaje Bluetooth retornado por el núcleo de ejecución indicando el resultado de la acción.

Si la acción fue exitosa, Domolution Mobile Link solicita a Domolution Control Core la actualización del estado del dispositivo que acaba de ser modificado.

Como punto final del proceso la notificación del resultado de la petición, tanto fallida como exitosa, es retornada al usuario por medio de un mensaje SMS, donde el subsistema Domolution Mobile será el encargado de interpretar el mensaje e informar el resultado.

##### 5) Domolution Execution Core

Este subsistema es el encargado de ejecutar las peticiones generadas por el usuario.

Se comunica con el subsistema Domolution Mobile Link, del cual recibe una petición transmitida por medio de un mensaje Bluetooth. Esta petición requiere que se ejecute la

modificación del estado de activación de un dispositivo, luego de lo cual se retorna un mensaje Bluetooth con el resultado de esta petición. La Fig. 4 muestra la comunicación entre Domolution Mobile Link y Domolution Execution Core.

Para efectos de este prototipo, los cambios de estados se realizan en una simulación de una casa domótica, la cual contiene diversos dispositivos a controlar.

La funcionalidad base de Domolution Execution Core fue escrita en lenguaje Java, apoyado por la librería de comunicación Bluetooth para Java Standard Edition Bluecove [12].

Éste núcleo de ejecución interactúa con la simulación desarrollada con la tecnología JavaFX [13], estando éste bajo de la capa de simulación y sobre la capa de comunicación de Bluecove.

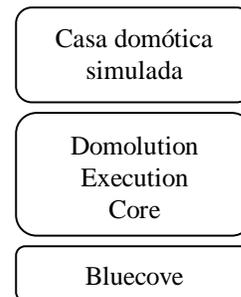


Fig. 6. Capas de interacción de Domolution Execution Core.

Bluecove es una implementación de JSR-82 [14] para la plataforma Java Standard Edition, que sirve de interfaz entre aplicaciones Java y la pila Bluetooth en diferentes sistemas operativos.

## IV. SEGURIDAD

Los subsistemas que interactúan directamente con el usuario manejan acceso restringido por nombre de usuario y contraseña.

Cuando se intenta tener acceso al subsistema Domolution Web, este es capaz de confirmar la identidad del usuario para luego solicitar conexión. Identificado el usuario se filtran las opciones asignadas para su tipo de usuario.

En el caso del acceso al subsistema Domolution Mobile, también es requerido un usuario y contraseña para poder acceder al sistema, pero la verificación de los datos es realizada por Domolution Control Core. Una vez confirmado se tiene acceso a los módulos del respectivo usuario.

### A. Third Party Verification (TPV)

La verificación de peticiones implementada en este trabajo sigue el modelo de servicio llamado **Third-Party Verification**, en adelante referido como **TPV**, el cual consiste en que un primer integrante solicita una petición a un segundo integrante, el cual consulta a un tercero, que conoce a los dos primeros, pidiendo verificar la identidad del primer integrante y sus privilegios asociados a la presente petición.

En éste contexto el primer integrante sería el sistema Domolution Mobile, el segundo integrante Domolution Mobile Link y el tercer integrante Domolution Control Core.

El proceso TPV se realiza basandose en los siguientes pasos:

1. Verifica si el dispositivo requerido existe.
2. Verifica si el dispositivo se encuentra bloqueado a los usuarios comunes.
3. Verifica si el estado de activación actual del dispositivo es el mismo que el deseado.
4. Verifica si el usuario posee los privilegios requeridos para manipular dicho dispositivo.

Si alguno de los cuatro pasos falla, el sistema Domolution Control Core retorna un mensaje a Domolution Mobile Link informando el paso que ha fallado. Este mensaje es retransmitido a Domolution Mobile, donde finalmente se muestra el mensaje al usuario.

Estos pasos del proceso TPV son importantes dado que los siguientes escenarios se pueden presentar al momento de solicitar la ejecución de una solicitud:

1. El dispositivo que el usuario desea manipular fue eliminado mientras el usuario generaba y transmitía la petición.
2. El dispositivo fue bloqueado mientras el usuario generaba y transmitía la petición y sus privilegios no le permiten controlar dispositivos bloqueados.
3. El estado de activación del dispositivo pudo haber sido modificado por un segundo usuario mientras el primero generaba y transmitía la petición.
4. Los privilegios del usuario pueden haber sido modificados mientras éste generaba y transmitía la petición.

Si se superan las cuatro verificaciones se dice que la solicitud puede ser ejecutada, y es entonces transmitida desde Domolution Mobile Link hacia Domolution Execution Core.

### B. Protocolo de comunicaciones

Con el objetivo de mantener una comunicación clara entre todos los componentes software del sistema Domolution durante la generación de peticiones, se diseñó una estructura de mensajes bajo la cual se construyen todos los mensajes transmitidos.

Los mensajes de petición transmitidos entre aplicaciones se conforman de las etiquetas DeviceCode, ActionCode y UserID como se muestra en la Fig. 7.



Fig. 7. Estructura de mensajes de petición

La etiqueta DeviceCode contiene el código del dispositivo sobre el cual se desea realizar la acción.

La etiqueta ActionCode contiene el código de la acción

deseada. Los códigos de acción se muestran en la Tabla I.

TABLA I  
ESTRUCTURA DE LOS MENSAJES DE PETICIÓN

Código de acción	Nombre de la acción
00	Apagar dispositivo
01	Encender dispositivo
10	Desbloquear dispositivo
11	Bloquear dispositivo

La etiqueta UserID contiene el número de identificación del usuario que genera la petición de la acción.

Los mensajes de respuesta son codificados con un único número, como lo muestra la Fig. 8.

### ResponseCode

1 Byte

Fig. 8. Estructura de los mensajes de respuesta.

Estas estructuras son utilizadas al generar peticiones de cambio de estado de dispositivos, al solicitar el proceso de TPV y al retornar las respectivas respuestas.

## V. CONCLUSIONES

La estructura de comunicaciones aquí planteada ha sido implementada en terminales hardware reales, pudiendo generar satisfactoriamente diversas peticiones las cuales son transferidas a través de las diversas redes y medios reales descritos a lo largo del documento, obteniendo un resultado satisfactorio en la simulación para cada petición.

Se propone la integración de esta arquitectura con un controlador hardware que reciba las peticiones Bluetooth y, a partir de la interpretación de éstas, ordene su ejecución en dispositivos electrónicos reales. De esta manera se puede prescindir de la simulación.

Al tener en cuenta el protocolo de mensajes propuesto se puede empaquetar una petición en un número reducido de Bytes. Gracias a esto, las transferencias de estas peticiones por medio de mensajes pequeños hacen que los procesos demanden tiempos y costos bajos.

## VI. AGRADECIMIENTOS

El autor reconoce las contribuciones de S. Pino y S. Castillo durante el diseño de la arquitectura distribuida.

Agradece de igual manera a S. Pino por las orientaciones impartidas a lo largo del desarrollo e implementación del prototipo.

Agradece también al Grupo de Investigación en Ingeniería Biomédica (GIIB) de la Universidad Industrial de Santander, por el apoyo recibido durante todo el proceso.

## VII. REFERENCIAS

- [1] *SMS tutorial. Developers Home*. (2009, Julio 28) [Online]. Available: <http://www.developershome.com/sms/>
- [2] B. Hopkins, R. Antony, *Bluetooth for Java*, cap. 1 y 2. Editorial Apress, 2003.
- [3] Sun Developer network, *The Java APIs for Bluetooth Wireless Technology* [Online]. Available: <http://developers.sun.com/mobility/midp/articles/bluetooth2/>
- [4] Y. Zhang, S. Zhang, S. Vuong, K. Malik, *Mobile Learning with Bluetooth-based E-learning System*, The University of British Columbia.
- [5] O. Henao, “*Hardware y Software Domótico. Estado del arte, historia de la domótica, aplicaciones en países extranjeros*”. Universidad Pontificia Bolivariana, 2006.
- [6] A. Tanenbaum. *Redes de computadoras*, 4a edición, Literal 1.1, 'Uso de las redes de computadora', Editorial Prentice-Hall inc., 2003.
- [7] *The Java EE 5* [Online]. Available: <http://java.sun.com/javaee/5/docs/tutorial/doc/docinfo.html>
- [8] S. Gálvez, L. Ortega, *Java a tope: J2ME (Java 2 Micro Edition)*, Cap. 3-4, 6-7, Universidad de Málaga.
- [9] A. Froufe, P. Cárdenas. *J2ME Java 2 Micro Edition, Cap. 1-4, 6-14 y 18*, Alfaomega, 2004.
- [10] B. Sarkar. *LWUIT 1.1 for Java ME Developers*, Cap. 1-11, Packt Publishing, 2009.
- [11] Sun Microsystems, *LWUIT Developers Guide, Lightweight UI Toolkit*, Cap. 1-12, Editado por Sun Microsystems, Inc.
- [12] *Bluecove, Java library for Bluetooth* [Online]. Available: <http://code.google.com/p/bluecove/>
- [13] Sun Developer Network, *Learning the JavaFX Script Programming Language - Tutorial Overview* [Online]. Available: <http://java.sun.com/javafx/1/tutorials/core/>
- [14] Java Community Process , *JSR 82: JavaTM APIs for Bluetooth* [Online]. Available: <http://jcp.org/en/jsr/detail?id=82>.
- [15] Java Community Process, “*JSR 118: Mobile Information Device Profile 2.0*” [Online]. Available: <http://jcp.org/en/jsr/detail?id=118>
- [16] Java Community Process, “*JSR 120: Wireless Messaging API*” [Online]: Available: <http://jcp.org/en/jsr/detail?id=120>
- [17] Java Community Process, Sun Microsystems, *Wireless Messaging API (WMA) for JavaTM 2 Micro Edition*, reference implementation. Version 1.0
- [18] The Apache Software Foundation, *Apache Struts*, [Online]. Available: <http://struts.apache.org/>
- [19] J. Elliot, “*Hibernate, A Developer's Notebook*”, Editorial O'REILLY, 2004.
- [20] J. Weaver, W. Gao, S. Chin, D. Iverson. “*Pro JavaFX Platform. Script, Desktop and Mobile RIA with JavaTM Technology*” in 'Adding Asynchronous Search with the JFXWorker', Editorial Apress, 2009.



**Carlos Gilberto Martínez Martínez** nació en Cartagena, Colombia, el 19 de Noviembre de 1985. Realizó sus estudios de educación superior en la Universidad Industrial de Santander, Bucaramanga, Colombia, donde obtuvo el título de Ingeniero de Sistemas en el año 2010. Entre sus campos de interés se encuentran las redes y telecomunicaciones y tecnologías móviles

Ejerció laboralmente como auxiliar docente en Vicerrectoría Académica de la Universidad Industrial de Santander durante su período universitario.

El Ingeniero Martínez obtuvo la distinción Cum Laude otorgada por la Universidad Industrial de Santander en el año 2010.